



TECHNISCHE UNIVERSITÄT MÜNCHEN

TUM School of Computation, Information and Technology

# Quantum Algorithms for Combinatorial Optimization

ALEXANDER KLIESCH

Vollständiger Abdruck der von der TUM School of Computation, Information and Technology der Technischen Universität München zur Erlangung des akademischen Grades eines

**Doktors der Naturwissenschaften** (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitz:

Prof. Dr. Felix Kraemer

Prüfer der Dissertation:

1. Prof. Dr. Robert König

2. Prof. Dr. Richard Kueng

Die Dissertation wurde am 25.11.2022 bei der Technischen Universität München eingereicht und durch die TUM School of Computation, Information and Technology am 03.07.2023 angenommen.



# Acknowledgments

First and foremost, I would like to thank Robert König, my doctoral advisor and supervisor. His guidance, passion and curiosity for science and research were a constant source of inspiration and motivation, and I could not have asked for a better person and researcher to guide me through my PhD.

Next, I would like to thank Michael Wolf, the head of M5, who has successfully managed to create a wonderful, productive, fun working atmosphere, and our secretary Silvia Schulz for taking care of administrative things and our many pleasant chats.

As this thesis is a cumulative one, the featured publications have been of utmost importance, and I would therefore like to thank my co-authors Sergey Bravyi, Libor Caha, Robert König, and Eugene Tang. It has been an honor working with them, and I consider myself lucky to have been able to learn from them.

In total, I have stayed at M5 in two stints, so I met many interesting, bright people that I thankfully can call both colleagues and friends. These people include my “PhD siblings” Beatriz Cardoso Diaz, Shin Ho Choe, Margret Heinze, and Stefan Huber, my office mates Francesco Battistel, Libor Caha, and Shangchun Yu, and all the others: Zahra Baghali Khanian, Vjosa Blakaj, Andreas Bluhm, Ángela Capel Cuevas, Matthias Caro, Xavier Coiteux-Roy, Diana Conache, Pablo Costa Rico, Javier Cuesta, Li Gao, Paul Gondolf, Martina Gschwendtner, Lisa Hänggli, Markus Hasenöhrl, Anna-Lena Hashagen, Yifan Jia, Haojian Li, Tristan Malleville, Chokri Manai, Tim Möbus, Ion Nechita, Emilio Onorati, Michael Prähofer, Hjalmar Rall, Silke Rolles, Cambyse Rouzé, Farzin Salek, Jeonghyeon Shin, Herbert Spohn, Daniel Stilck-França, Quirin Vogel, Simone Warzel, and Amanda Young. All of them have made my time at M5 a wonderful experience that I will forever cherish.

I would also like to thank my PhD mentor Andreas Johann who fortunately never had to interfere in any unpleasant matters (there simply weren’t any of those), Lydia Weber from the Dean’s office and ISAM coordinator Isabella Wiegand who both helped me with the organisational parts of my PhD (especially at the end), and my therapist Martina Beck, whose helpful, continuing support has helped tremendously and is very much appreciated.

I would also like to thank my close friends Rufat Badal, Bernhard Blieninger, Vincent Karbassioun, Maximilian Schiller, Dominik Stöger, Christoph Striegel, and Andreas Wasmeier. Last, but not least, I would like to thank my family: my parents Brigitte and Helmut Kliesch, my uncles Johann and Ludwig Rasch, and my wonderful sisters Elke, Marion, and Christina Kliesch.



*“ I’d argue that everybody wants to do something that matters. ”*

—— LINUS TORVALDS



# Zusammenfassung

Diese Dissertation untersucht variationelle Quanten-Algorithmen, welche wir dazu verwenden, NP-vollständige kombinatorische Optimierungsprobleme wie das MAX-CUT-Problem approximativ zu lösen. Unser Hauptaugenmerk liegt auf dem von Farhi, Goldstone, and Gutmann eingeführten “Quantum Approximate Optimization Algorithm” (QAOA), welcher aufgrund seiner potentiellen zeitnahen Anwendung auf echten Quanten-Computern erhebliches Interesse auf sich gezogen hat. Insbesondere beweisen wir Leistungsgrenzen von QAOA. Darüber hinaus schlagen wir zwei neue, von QAOA abgeleitete Algorithmen vor, welche darauf abzielen, diese Beschränkungen zu umgehen: einer dieser Algorithmen, welchen wir “Recursive QAOA” nennen, benutzt QAOA rekursiv als eine Subroutine, und der zweite Algorithmus, welchen wir “twisted QAOA” nennen, macht sich Post-Processing zu nutze.





# Abstract

This thesis studies variational quantum algorithms applied to approximately solve NP-complete combinatorial optimization problems such as the MAX-CUT problem. Our main focus lies on the “Quantum Approximate Optimization Algorithm” (QAOA) introduced by Farhi, Goldstone, and Gutmann which has attracted considerable attention due to its potential applicability on near-term real-world quantum devices. In particular, we prove performance limitations of QAOA. Furthermore, we propose two new algorithms derived from QAOA which aim at bypassing these limits: an algorithm using QAOA as a subroutine called “recursive QAOA” and a version incorporating post-processing called “twisted QAOA”.



# List of contributed articles

## *Core articles as principal author*

- I) Sergey Bravyi, Alexander Kliesch, Robert Koenig, and Eugene Tang  
“Hybrid quantum-classical algorithms for approximate graph coloring”  
*Quantum* 6, 678 (2022)  
(see also article [1] in the bibliography)
- II) Libor Caha, Alexander Kliesch, and Robert Koenig  
“Twisted hybrid algorithms for combinatorial optimization”  
*Quantum Science and Technology* 7, 045013 (2022)  
(see also article [2] in the bibliography)

## *Articles as co-author*

- III) Sergey Bravyi, Alexander Kliesch, Robert Koenig, and Eugene Tang  
“Obstacles to Variational Quantum Optimization from Symmetry Protection”  
*Physical Review Letters* 125, 260505 (2020)  
(see also article [3] in the bibliography)

I, Alexander Kliesch, confirm that I am the principal author of the core articles of this dissertation which are I) and II).



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Summary of results . . . . .	2
1.2	Outline . . . . .	4
<b>2</b>	<b>Preliminaries</b>	<b>5</b>
2.1	Classical Computation . . . . .	5
2.1.1	Computational Problems . . . . .	5
2.1.2	Algorithms and Turing machines . . . . .	8
2.1.3	Complexity of computational problems . . . . .	9
2.1.4	Approximation Algorithms . . . . .	11
2.2	Quantum Physics . . . . .	15
2.2.1	Hilbert space essentials . . . . .	15
2.2.2	Operator terminology . . . . .	17
2.2.3	Dirac notation . . . . .	17
2.2.4	The postulates of quantum physics . . . . .	18
2.2.5	Further remarks . . . . .	19
2.3	Quantum Computation . . . . .	20
2.3.1	Quantum Circuits . . . . .	21
2.3.2	Quantum Algorithms . . . . .	23
<b>3</b>	<b>The Quantum Approximate Optimization Algorithm (QAOA)</b>	<b>25</b>
3.1	Definition of QAOA . . . . .	25
3.1.1	The Quantum Adiabatic Algorithm (QAA) . . . . .	25
3.1.2	Examples for encodings of combinatorial optimization problems . . . . .	26
3.1.3	From QAA to the QAOA ansatz . . . . .	27
3.1.4	Using QAOA for combinatorial optimization . . . . .	29
3.2	Properties of the QAOA ansatz . . . . .	32
3.2.1	Locality . . . . .	32
3.2.2	Uniformity . . . . .	33
3.2.3	Circuit Depth for two-local Hamiltonians . . . . .	34
3.3	Performance guarantees for QAOA . . . . .	34

## CONTENTS

<b>4</b>	<b>Performance limitations of QAOA</b>	<b>35</b>
4.1	Limitations obtained by exploiting locality and symmetry . . . . .	35
4.2	Limitations obtained by exploiting locality and uniformity . . . . .	37
4.3	Further known limitations of QAOA in the literature . . . . .	38
<b>5</b>	<b>Enhancements of QAOA</b>	<b>39</b>
5.1	Recursive QAOA . . . . .	39
5.2	Twisted QAOA . . . . .	42
5.3	Other proposals for enhancing QAOA in the literature . . . . .	44
	<b>Bibliography</b>	<b>45</b>
	<b>Appendices</b>	
<b>A</b>	<b>Core Articles</b>	<b>51</b>
A.1	Hybrid quantum-classical algorithms for approximate graph coloring . . . . .	51
A.1.1	Main Results . . . . .	52
A.1.2	Individual Contribution . . . . .	52
A.2	Twisted hybrid algorithms for combinatorial optimization . . . . .	81
A.2.1	Main Results . . . . .	82
A.2.2	Individual Contribution . . . . .	82
<b>B</b>	<b>Article as co-author</b>	<b>105</b>
B.1	Obstacles to Variational Quantum Optimization from Symmetry Protection . . .	105
B.1.1	Main Results . . . . .	106
B.1.2	Individual Contribution . . . . .	106

# Chapter 1

## Introduction

The field of quantum computation, first envisioned in the early 1980s [4, 5], is currently in a transitional phase in its development. On one hand, the capabilities of real-world devices are continuously increasing as evidenced by the celebrated claims of quantum advantage by Google [6] and subsequent introduction of devices capable of harnessing even more computational power [7] as quantified by the number of fundamental quantum information units (generally called “qudits” and more specifically “qubits” when in two-level form). On the other hand, the number of such qubits in current devices is still not large enough to realize promising applications of algorithms such as Shor’s factoring algorithm [8] and Grover’s search algorithm [9] due to the detrimental effects of quantum decoherence. While the potential errors occurring in a quantum computation, e.g., by decoherence or faulty gates and measurements, can be overcome in principle [10], the advancements necessary to achieve scalable universal fault-tolerant quantum computation in the lab are sufficiently challenging that this is not expected to be realizable in the near future.

This transitional phase has been called the “NISQ era” [11], where the abbreviation stands for “Noisy Intermediate-Scale Quantum”. In the absence of universal fault-tolerant quantum computation, the goal is instead to focus on quantum algorithms whose physical realizations as quantum circuits take very little time such that the negative effects of decoherence are at least reduced. Such algorithms are realized via so-called short-depth quantum circuits; they are the focus of this thesis. Informally, a quantum circuit satisfies this requirement if the total time that the quantum computation takes to finish all of its actions (i.e., its depth) does not grow too strongly (e.g., at most logarithmically) with the size of the problem instance.

While quantum processing capabilities are limited at the current time, classical computational resources are powerful and ubiquitous. Therefore, a feasible approach to enhance the capabilities of short-depth quantum circuits is to supplement the quantum computation by classical computation. One way of doing this is by letting the quantum circuit depend on a set of real parameters which are found via classical means. The analytical study of the performance of algorithms of this type are the overarching topic of this thesis; they are called variational quantum algorithms. In particular, we will focus on one specific variational quantum algorithm applied to combinatorial optimization problems, the Quantum Approximate Optimization Algorithm, abbreviated as “QAOA” [12].

This algorithm is a so-called approximation algorithm, i.e., an efficient algorithm that outputs an approximate solution to a combinatorial optimization problem (specified via a given cost function whose value we wish to optimize), meaning that the obtained output is typically not optimal. Such algorithms are motivated by the fact that many combinatorial optimization problems are expected to be computationally hard to solve exactly. To measure the performance of approximation algorithms, we will use the so-called approximation ratio. This figure of merit is defined as the (expected) value of the cost function applied to the output of the algorithm divided by the actual optimum of the problem at hand; it is therefore always a quantity between 0 and 1 for maximization problems.

To use QAOA, one chooses a natural number called the “level” (usually denoted by  $p$ ) before applying the algorithm which we then denote by  $\text{QAOA}_p$ . While increasing  $p$  improves the performance of the algorithm, it also increases the depth of the quantum circuit and therefore makes it more susceptible to the effects of decoherence. Furthermore, the number of real parameters in the quantum circuit (which need to be determined via classical means) is equal to  $2p$  and therefore also determined by the level; both these  $p$ -dependent properties motivate a choice of small  $p$ .

In this thesis, we focus on finding both performance guarantees and limitations of  $\text{QAOA}_p$  depending on the level  $p$ . In particular, we compare the performance of QAOA with the performance of established classical algorithms. To overcome any limitations that we observe, we propose enhancements of this algorithm which we hope might perform better and investigate their performance as well.

## 1.1 Summary of results

We give an overview of the published research articles which constitute the backbone of this thesis, where Article I) [1] and II) [2] are the core articles for which the author of this thesis is the principal author, while the author does not claim principal authorship for Article III) [3].

### *Core articles as principal author*

- Article I) [1]: *Hybrid quantum-classical algorithms for approximate graph coloring*

This core article has three main contributions to the study of variational quantum algorithms. First, we apply techniques similar to the ones previously used in [13] for the MAX-CUT problem to establish worst-case bounds on the performance of QAOA applied to the MAX- $k$ -CUT problem, where  $k \geq 2$  is a constant integer. In this combinatorial optimization problem (which is intimately connected to the well-known graph coloring problem), one attempts to color the vertices of a given graph using  $k$  colors such that as many edges as possible connect vertices with different colors; choosing  $k = 2$  yields the MAX-CUT problem. Our result implies that if one chooses the level of QAOA as a constant, i.e., independently of the number of vertices in the graph, the performance of the algorithm applied to almost all  $d$ -regular bipartite graphs with  $n$  vertices, where  $n$  and  $d$  are large and suitably chosen, is comparable to simply randomly guessing colors.



A crucial part of QAOA<sub>p</sub> is the choice of  $2p$  real parameters. Typically, these parameters are chosen such that the expected value of the so-called problem Hamiltonian in the input state is maximized. Usually, this expectation value consists of a sum of terms of the form  $\langle \psi(\beta, \gamma) | Z_w^r Z_v^s | \psi(\beta, \gamma) \rangle$ , where  $|\psi(\beta, \gamma)\rangle$  is the output state of QAOA<sub>p</sub> with parameters  $\beta, \gamma \in \mathbb{R}^p$  and  $Z_w^t$  is the  $t$ -th power of the generalized Pauli- $Z$  operator applied to subsystem  $w$ . Being able to determine these quantities is important as the approximation ratio is defined as a function of such expectation values. Therefore, as a second contribution, we introduce a classical algorithm whose runtime on bounded-degree graphs scales linearly in the number of vertices of the given graph and computes this expectation value for the case  $p = 1$ . This algorithm can also be used to numerically study the performance of RQAOA<sub>1</sub>, an algorithm introduced in Article III) [3], which recursively uses QAOA<sub>1</sub> as a subroutine. Third, we use this classical algorithm to compare the performance of three algorithms applied to the MAX-3-CUT problem in terms of their approximation ratios, namely QAOA<sub>1</sub>, RQAOA<sub>1</sub>, and the best classical algorithm for the problem given by Newman [14]. These algorithms are applied to a variety of different graph ensembles. We see empirically that RQAOA<sub>1</sub> consistently outperforms QAOA<sub>1</sub> for all ensembles, and moreover, for certain ensembles, RQAOA<sub>1</sub> is able to outperform Newman’s algorithm. However, there are also certain regimes for which the classical algorithm outperforms RQAOA<sub>1</sub>.

- Article II) [2]: *Twisted hybrid algorithms for combinatorial optimization*

In the second core article, we use a different approach to enhance the capabilities of QAOA applied to the MAX-CUT problem by performing classical post-processing after the application of a variational quantum algorithm. Inspired by the well-established techniques for classical algorithms of Feige, Karpinski, and Langberg [15] and Halperin, Livnat, and Zwick [16] which are suitable for the MAX-CUT problem on 3-regular graphs, we use the original level- $p$  QAOA circuit family, but choose the parameters to maximize an altered cost function that already takes the subsequent post-processing into account. This change (or “twist”, hence the name of the algorithm) of the cost function is motivated by the availability of provable improvement guarantees for both of these techniques.

Expressing these guarantees as expectation values of local Hamiltonians and exploiting the well-known locality of QAOA, we are able to derive provable, analytical lower bounds on the approximation ratio of twisted QAOA applied to MAX-CUT on 3-regular graphs for low levels of  $p$ . We see that twisted QAOA noticeably improves upon “bare” QAOA of the same level while using the exact same amount of quantum resources. In fact, the obtained results suggest that the performance of twisted QAOA with level  $p$  is comparable to the performance of bare QAOA with level  $p + 1$ , meaning that using the twisted algorithm decreases the total depth of the quantum circuit in comparison to the original algorithm.

#### *Further article*

- Article III) [3]: *Obstacles to Variational Quantum Optimization from Symmetry Protection*

In this article, we exploit the symmetry properties of QAOA to establish upper bounds on the performance (as quantified by the approximation ratio) of low-level QAOA applied

to the MAX-CUT problem; in particular, our results hold for constant-level QAOA. Our result is motivated by the observation that both the QAOA circuit family for the MAX-CUT problem and the input state of QAOA obey a special symmetry property with respect to the operator  $X^{\otimes n}$ , where  $X$  is the Pauli- $X$  operator and  $n$  is the input size of the circuit; we call this property  $\mathbb{Z}_2$ -symmetry. Consider now a family of MAX-CUT problem Hamiltonians associated with a particularly chosen, infinite ensemble of graphs. We then show that there is a positive constant  $\varepsilon$  and a natural number  $n_0$  such that if  $n \geq n_0$ , any  $n$ -qubit quantum state that has an energy density lower than  $\varepsilon$  cannot be prepared purely by applying a short-depth  $\mathbb{Z}_2$ -symmetric quantum circuit to a  $\mathbb{Z}_2$ -symmetric input product state.

Using this result, we then establish upper bounds on the performance of logarithmic-depth QAOA applied to MAX-CUT on so-called Ramanujan graphs. In particular, the worst-case approximation ratio of QAOA is strictly worse than the corresponding approximation ratio achieved by the classical Goemans-Williamson algorithm [17]. This implies that constant-level QAOA cannot achieve a higher approximation ratio than the best classical algorithms in the worst case.

Furthermore, we introduce a recursive version of QAOA called “RQAOA” which recursively uses QAOA to determine energetic (anti-)correlations between vertices in a given graph to successively decrease the number of vertices, allowing a direct classical solution followed by a reconstruction of a solution candidate of the original problem. We show that for the Ising model defined on cycle graphs, level-1-RQAOA outperforms QAOA of any level. We subsequently study the application of this algorithm for the MAX-3-CUT problem in Core Article I) [1].

## 1.2 Outline

In Chapter 2, we review some basics from classical computer science and quantum information theory. In Chapter 3, we give a detailed introduction to the main topic of this thesis, the Quantum Approximate Optimization Algorithm (QAOA). In Chapter 4, we present our results on limitations of QAOA which – using different techniques – imply that the worst-case performance of QAOA is insufficient, especially compared to classical algorithms. Lastly, in Chapter 5, we give two proposals to enhance the QAOA ansatz by using a recursive and a post-processed version of QAOA, respectively.

# Chapter 2

## Preliminaries

This chapter provides some background information necessary to understand the content of this thesis. Conceptually, the topics addressed in this thesis lie at the intersection of computer science and quantum physics, leading to quantum computation.

Let us first fix some mathematical notation that we will use. In this thesis,  $\mathbb{N} = \{1, 2, 3, \dots\}$  denotes the set of *positive* natural numbers, and we define  $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$ . For  $n \in \mathbb{N}$ , we set  $[n] := \{1, \dots, n\}$  and use the expression  $\mathbb{Z}_n$  to denote both the set  $\{0, \dots, n-1\}$  as well as the additive cyclic group with  $n$  elements. Furthermore, we set  $\mathbb{R}_{>0} := (0, \infty)$  and  $\mathbb{R}_{\geq 0} := [0, \infty)$ .

For any set  $A \subset \mathbb{N}$ , we denote the set of all ordered tuples containing  $m$  elements of  $A$  by  $A^m$ , the set of all subsets of  $A$  with exactly  $m$  elements by  $\binom{A}{m}$ , and the set of all finite-length strings consisting of elements of  $A$  by  $A^*$ . For  $x \in A^*$ , we denote its length by  $|x|$  and the  $\ell$ -th entry of  $x$  by  $x_\ell$ . Given two strings  $x, y \in A^*$ , we denote the concatenation of  $x$  and  $y$  by  $xy := x_1 \dots x_{|x|} y_1 \dots y_{|y|}$ , and if  $0 \in A$ , we denote the Hamming weight of  $x$ , i.e., the number of nonzero entries in  $x$ , by  $|x|_H$ .

### 2.1 Classical Computation

Classical computer science is the formal study of classical computers and computational systems. For a more extensive introduction to the relevant subfields of this thesis, see [18] and [19].

#### 2.1.1 Computational Problems

In this thesis, we will concern ourselves with the study of computational problems.

**Definition 2.1** (Computational Problem)

A **computational problem** is a triplet  $(\mathcal{I}, \mathcal{F} = \{\mathcal{F}(I)\}_{I \in \mathcal{I}}, \mathcal{S} = \{\mathcal{S}(I)\}_{I \in \mathcal{I}})$ , where

- (i)  $\mathcal{I}$  is the **instance set**,
- (ii) for  $I \in \mathcal{I}$ ,  $\mathcal{F}(I)$  is the **feasible set** associated with  $I$ , and
- (iii) for  $I \in \mathcal{I}$ ,  $\mathcal{S}(I) \subseteq \mathcal{F}(I)$  is the **solution set** associated with  $I$ .

Given  $I \in \mathcal{I}$ , the goal is to determine an element of  $\mathcal{S}(I)$ .

Note that in this definition, there is freedom in choosing the feasible sets; adding elements to any of them does not change the solution sets and therefore does not change the result of the operational task at hand. In spirit, feasible sets can be thought of as the set of solution “guesses” that the person posing the computational problem would deem reasonable, even if they are not actual solutions to the problem.

One of the most basic computational problems models a Yes-or-No question. Recall that a set  $L \subseteq \{0, 1\}^*$  is called a **language**.

**Definition 2.2** (Decision Problem)

Let  $L$  be a language. The **decision problem associated with  $L$**  is the computational problem  $(\{0, 1\}^*, \{\mathcal{F}(x)\}_{x \in \{0, 1\}^*}, \{\mathcal{S}(x)\}_{x \in \{0, 1\}^*})$ , where  $\mathcal{F}(x) = \{0, 1\}$  for all  $x \in \{0, 1\}^*$  and

$$\mathcal{S}(x) = \begin{cases} \{1\} & \text{for } x \in L \quad , \\ \{0\} & \text{for } x \notin L \quad . \end{cases}$$

For example, if we set  $L$  to be the set of all prime numbers represented by binary strings, the associated decision problem models the task of determining whether a given binary string represents a prime number.

For most of this thesis, however, we will focus on combinatorial optimization problems.

**Definition 2.3** (Combinatorial Optimization Problem)

A **combinatorial optimization problem** is a quadruple

$$(\mathcal{I}, \mathcal{F} = \{\mathcal{F}(I)\}_{I \in \mathcal{I}}, \mathcal{S} = \{\mathcal{S}(I)\}_{I \in \mathcal{I}}, c = \{c_I\}_{I \in \mathcal{I}})$$

such that

- (i)  $(\mathcal{I}, \mathcal{F} = \{\mathcal{F}(I)\}_{I \in \mathcal{I}}, \mathcal{S} = \{\mathcal{S}(I)\}_{I \in \mathcal{I}})$  is a computational problem,
- (ii) for all  $I \in \mathcal{I}$ ,  $|\mathcal{F}(I)| < \infty$ ,
- (iii) for all  $I \in \mathcal{I}$ ,  $c_I$  is a function  $c_I: \mathcal{F}(I) \rightarrow \mathbb{R}$ , and
- (iv) for all  $I \in \mathcal{I}$ , we have

$$\mathcal{S}(I) = \{z \in \mathcal{F}(I): c_I(z) = \max_{y \in \mathcal{F}(I)} c_I(y)\} .$$

We call  $c$  the **objective function** associated with the combinatorial optimization problem.

Note that we assume that every combinatorial optimization problem is a maximization problem. This can be done without loss of generality since minimization of a real-valued function  $f$  is equivalent to maximization of  $-f$ .

Combinatorial optimization problems have a wide range of practical applications. Let us give examples for some of these problems, with the first two based on **graphs**. Recall that an **undirected, unweighted graph**  $G = (V, E)$  is specified via its **vertex set**  $V$  and its **edge**

set  $E \subseteq \binom{V}{2}$ , whereas an **undirected, weighted graph**  $G = (V, E, w)$  is additionally equipped with a **weight function**  $w: E \rightarrow \mathbb{R}$ . Any unweighted graph can be interpreted as a weighted graph by setting  $w(e) = 1$  for all  $e \in E$ . We call a  $k$ -tuple  $p = (p_1, \dots, p_k) \in E^k$  a **path** of length  $k$  if for all  $\ell \in [k-1]$ , we have  $|p_\ell \cap p_{\ell+1}| \geq 1$ ; if additionally  $|p_1 \cap p_k| \geq 1$  holds, we call  $p$  a **cycle** of length  $k$ .

1. For the definition of the **traveling salesperson problem** (TSP), recall that a graph  $G = (V, E)$  is called complete if  $E = \binom{V}{2}$ . Furthermore, a cycle  $h$  of length  $|V|$  is called a Hamiltonian cycle of  $G$  if every  $u \in V$  is contained in exactly two entries of  $h$ . Then the TSP is specified via

$$\begin{aligned} \mathcal{I} &:= \{G: G = (V, E, w) \text{ complete, undirected, weighted graph}\} \quad , \\ \text{for } G \in \mathcal{I}, \quad \mathcal{F}(G) &:= \{h: h \text{ is Hamiltonian cycle in } G\} \quad , \\ c_G: \mathcal{F}(G) &\rightarrow \mathbb{R} \quad , \\ h &\mapsto - \sum_{k \in \mathbb{Z}_{|V|}} w(h_k) \quad . \end{aligned}$$

The most commonly used real-world formulation of TSP is given via a traveler who starts their journey in their hometown, wants to visit all cities on a given list exactly once before returning home and asks which route they should take to minimize the total distance traveled.

2. Let  $k \geq 2$  be an integer. Then the unweighted **MAX- $k$ -CUT problem** is a combinatorial optimization problem with

$$\begin{aligned} \mathcal{I} &:= \{G: G = (V, E) \text{ undirected, unweighted graph}\} \quad , \quad (2.1) \\ \text{for } G \in \mathcal{I}, \quad \mathcal{F}(G) &:= \mathbb{Z}_k^{|V|} \quad , \\ c_G: \mathcal{F}(G) &\rightarrow \mathbb{R} \quad , \\ x &\mapsto \sum_{\{u,v\} \in E} (1 - \delta_{x_u, x_v}) \quad , \end{aligned}$$

where  $\delta_{ab}$  is the **Kronecker delta**, i.e., it evaluates to 1 if  $a = b$  and 0 else.

In other words, the task in the unweighted MAX- $k$ -CUT problem is to find an assignment of the vertices of a given graph to the set  $\mathbb{Z}_k$  (a so-called “**cut**”) such that the cardinality of the set of edges that connect two vertices with different assigned values is maximized. In the following, such edges will be called **satisfied**, and given a cut  $C$  of a graph  $G$ , we will call the number of satisfied edges in  $C$  the **cutsize** of  $C$  in  $G$  and denote it by  $\text{cutsize}_G(C)$ . A cut can be either specified via a function  $C: V \rightarrow \mathbb{Z}_k$  or a string  $C \in \mathbb{Z}_k^{|V|}$ ; we will use both specifications interchangeably in the following. Note that there is also a weighted version of the MAX- $k$ -CUT problem, where each summand in the definition of  $c_G$  is multiplied by the corresponding edge weight; the goal there is to maximize the total weight of satisfied edges.

If we set  $k = 2$ , we obtain the unweighted **MAX-CUT problem**.

3. The **0-1 knapsack problem** is specified via

$$\begin{aligned} \mathcal{I} &:= \{(w, v, W) : W \in \mathbb{R} \text{ and } \exists n \in \mathbb{N} \text{ s. t. } w, v \in \mathbb{R}^n\} \quad , \\ \text{for } (w, v, W) \in \mathcal{I}, \quad \mathcal{F}(w, v, W) &:= \{x \in \{0, 1\}^{|w|} : \sum_{k=1}^{|w|} w_k x_k \leq W\} \quad , \\ c_{(w, v, W)} : \mathcal{F}(w, v, W) &\rightarrow \mathbb{R} \quad , \\ x \mapsto \sum_{k=1}^{|w|} v_k x_k &\quad . \end{aligned}$$

One can interpret this problem as being given a container (such as a backpack) which has a maximum weight capacity  $W$  together with a list of  $n$  items which have both a weight (modeled by  $w$ ) and a value (modeled by  $v$ ). The task is then to decide which items should be put into the container such that it contains the maximum possible value while not exceeding the given maximum weight.

### 2.1.2 Algorithms and Turing machines

The question of how to solve computational problems concretely leads to the study of algorithms. Loosely speaking, an **algorithm**  $\mathcal{A}$  for a computational problem  $(\mathcal{I}, \mathcal{F}, \mathcal{S})$  is a sequence of unambiguous instructions that, upon inputting  $x \in \mathcal{I}$ , outputs an element  $y = \mathcal{A}(x) \in \mathcal{F}(x)$ , and we say that  $\mathcal{A}$  **solves the computational problem** if  $\mathcal{A}(x) \in \mathcal{S}(x)$  for every  $x \in \mathcal{I}$ . While it is well known that some computational problems such as the halting problem [20] are **uncomputable**, i.e., there is no algorithm that solves it, all the computational problems in this thesis will admit such an algorithm.

To formalize the above notion of an algorithm, several models of computation have been proposed. The two most commonly used models are the **Turing machine model** and the **circuit model**. To rigorously define computation and computational complexity, we will exclusively work within the Turing machine model in the classical setting while in the quantum algorithmic setting, we will exclusively work within the circuit model as it is easier to work with in practice. Our description of the Turing machine model will be rather brief; we refer to [18] for a more extensive treatment.

A (deterministic) **Turing machine**  $M$  is specified by a triplet  $(\Gamma, Q, \delta)$ , where  $\Gamma \supseteq \{0, 1, \square, \triangleright\}$  (the **alphabet**) and  $Q \supseteq \{q_{\text{start}}, q_{\text{halt}}\}$  (the **state set**) are two finite sets and  $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{-1, 0, 1\}$  is called the **transition function**. The machine consists of a **tape**, i.e., an infinite one-directional line of **cells**, where each cell holds one element of  $\Gamma$ , a **tape head** that is pointing at one cell and is capable of reading and/or replacing the content of the cell that it is currently pointing at with elements from  $\Gamma$ , and a **register** which holds one element of  $Q$ . We call a tuple  $(q, k, t)$  a **configuration of**  $M$ , where  $q$  is the content of the register,  $k \in \mathbb{N}$  represents the position of the tape head, and  $t = t_1 t_2 \dots$  is the content of the tape, where  $t_\ell \in \Gamma$  for all  $\ell \in \mathbb{N}$ .

To apply an algorithm specified by a given Turing machine to the binary input  $x \in \{0, 1\}^*$ , one first initializes the machine in the configuration  $(q_{\text{start}}, 1, \triangleright x \square \square \dots)$ . Subsequently, the transi-

tion function is repeatedly applied until the register contains the state  $q_{\text{halt}}$ ; at this point, the **output**  $M(x)$  is obtained by reading out the content of the tape and disregarding any  $\square$  or  $\triangleright$ . If the configuration of the Turing machine before an application of  $\delta$  was  $(q, k, t)$ , the configuration after an application is  $(\delta_1(q, t_k), \max\{1, k + \delta_3(q, t_k)\}, t_1 \dots t_{k-1} \delta_2(q, t_k) t_{k+1} \dots)$ , where for  $\ell \in \{1, 2, 3\}$ , the expression  $\delta_\ell(q, t_k)$  denotes the  $\ell$ -th component of  $\delta(q, t_k)$  and the third component of the configuration should be interpreted in the sense of string concatenation.

Extensions of the model described above exist. For example, one could define Turing machines with several tapes, tapes which are infinite in both directions, or two-dimensional tapes. Moreover, the Turing machine model introduced above is **deterministic**, i.e., feeding the same input to the Turing machine will always produce the same output, a property that is not shared by so-called **probabilistic Turing machines**. These devices are specified by a quadruple  $(\Gamma, Q, \delta^1, \delta^2)$ , where both  $\delta^1$  and  $\delta^2$  are transition functions as described in the deterministic setting. The fundamental difference between deterministic and probabilistic Turing machines is that during each computational step in the probabilistic version, one randomly picks one of the transition functions and applies this function. Probabilistic Turing machines capture the notion of **probabilistic** (also called **randomized**) **algorithms** which utilize randomness: in general, if  $\delta^1 \neq \delta^2$ , one run of the algorithm might produce an output that differs from the output of another run.

### 2.1.3 Complexity of computational problems

Given two algorithms that solve the same computational problem, one can wonder whether one of the algorithms is better suited for the task than the other according to some criterion, e.g., time or space needed to perform the algorithm. In general, we are not interested in an exact expression for such quantities; instead, we care about the behaviour for large input sizes and therefore use asymptotic notation which we briefly recall here.

**Definition 2.4** (Asymptotic Notation [18])

Let  $f, g: \mathbb{N} \rightarrow \mathbb{R}_{\geq 0}$  be two functions.

(i) We say that  $f = O(g)$  if there exist a constant  $c$  and  $n_0 \in \mathbb{N}_0$  such that

$$f(n) \leq c \cdot g(n) \quad \text{for all } n \geq n_0 \quad .$$

(ii) We say that  $f = \Omega(g)$  if  $g = O(f)$ .

(iii) We say that  $f = \Theta(g)$  if  $f = O(g)$  and  $g = O(f)$ .

(iv) We say that  $f = o(g)$  if for every  $\varepsilon > 0$ , there exists  $n_0 \in \mathbb{N}_0$  such that

$$f(n) \leq \varepsilon \cdot g(n) \quad \text{for all } n \geq n_0 \quad .$$

(v) We say that  $f = \omega(g)$  if  $g = o(f)$ .

We will follow the convention of including the input parameter  $n$  in the statement, e.g., we write  $f(n) = O(g(n))$  instead of  $f = O(g)$ .

In terms of classical computation, we will only be interested in time as a resource, where we measure time in terms of the number of applications of transition functions. We call a single application of a transition function a **computational step**.

**Definition 2.5** (Deterministic Time Complexity)

Let  $M = (\Gamma, Q, \delta)$  be a deterministic Turing machine and  $T: \mathbb{N} \rightarrow \mathbb{N}$ .

- (i) We say that  $M$  **runs in time**  $T$  if for every  $x \in \{0, 1\}^*$ , the register of the machine is in the state  $q_{halt}$  after at most  $T(|x|)$  applications of  $\delta$ .
- (ii) We say that  $M$  is a **deterministic polynomial-time algorithm** if there is a polynomial  $T$  such that  $M$  runs in time  $T$ .

We can now define the concept of **complexity classes**. Roughly speaking, a complexity class is a set of computational problems that share certain features about the algorithms that solve them. The most fundamental complexity classes concerning runtimes are captured in the following definition.

**Definition 2.6** (P, NP, Polynomial-Time Reducibility, NP-hard and NPC)

Let  $L$  be a language.

- (i) We say that  $L \in \mathbf{P}$  if there exists a deterministic polynomial-time algorithm that solves the decision problem associated with  $L$ .
- (ii) We say that  $L \in \mathbf{NP}$  if there exists a deterministic polynomial-time algorithm  $M$  such that for all  $x \in L$ , there exists  $y \in \{0, 1\}^*$  such that  $M(x, y) = 1$ .
- (iii) Let  $\tilde{L} \in \mathbf{NP}$ . We say that  $\tilde{L}$  is (**polynomial-time**) **reducible** to  $L$  if there exists a deterministic polynomial-time algorithm  $M$  such that, given input  $x \in \{0, 1\}^*$ ,

$$x \in \tilde{L} \Leftrightarrow M(x) \in L \quad .$$

- (iv) We say that  $L$  is **NP-hard** if every language  $\tilde{L} \in \mathbf{NP}$  is reducible to  $L$ .
- (v) We call  $L$  **NP-complete** if  $L$  is NP-hard and  $L \in \mathbf{NP}$ . The complexity class of all NP-complete languages is denoted by **NPC**.

By definition, it is clear that  $\mathbf{P} \subseteq \mathbf{NP}$ . However, it is not known whether  $\mathbf{P}$  is in fact equal to  $\mathbf{NP}$ ; this question is also known as the **P-NP-problem** and is considered to be one of the most fundamental open problems of theoretical computer science. It is clear from the definition that  $\mathbf{P} \cap \mathbf{NPC} \neq \emptyset$  if and only if  $\mathbf{P} = \mathbf{NP}$ . This implication illustrates the importance of NP-complete problems as the “hardest” problems within NP.

While combinatorial optimization problems themselves cannot be elements of  $\mathbf{P}$  or  $\mathbf{NP}$  by definition since they are not decision problems, we can transform them to equivalent decision problems.



For example, the unweighted MAX-CUT problem as defined in (2.1) with  $k = 2$  can be formulated as a decision problem associated with the following language:

$$L_{\text{MAX-CUT}} := \{(G, \ell) : \ell \in \mathbb{Z} \text{ and } G = (V, E) \text{ undirected, unweighted graph such that } \max_{x \in \{0,1\}^{|V|}} c_G(x) \geq \ell\} \quad ,$$

where we assume that the elements of  $L_{\text{MAX-CUT}}$  are encoded as binary strings. In their decision versions, the combinatorial optimization problems introduced in Section 2.1.1, i.e., TSP, MAX-CUT, MAX- $k$ -CUT for any integer  $k \geq 3$ , and the 0 – 1 knapsack problem, all turn out to be NP-complete [21], and many other NP-complete combinatorial optimization problems are known. One can also define complexity classes using randomized algorithms. To do so, we extend the concept of runtime to the probabilistic setting.

**Definition 2.7** (Probabilistic Time Complexity)

Let  $M = (\Gamma, Q, \delta^1, \delta^2)$  be a probabilistic Turing machine and  $T: \mathbb{N} \rightarrow \mathbb{N}$ .

- (i) We say that  $M$  **runs in time**  $T$  if for every  $x \in \{0,1\}^*$ , the register of the machine is in the state  $q_{\text{halt}}$  after at most  $T(|x|)$  applications of either  $\delta^1$  or  $\delta^2$ .
- (ii) We say that  $M$  is a **probabilistic polynomial-time algorithm** if there is a polynomial  $T$  such that  $M$  runs in time  $T$ .

Given a specific run of a probabilistic Turing machine  $M$  applied to input  $x \in \{0,1\}^*$ , we call the sequence of applications of either  $\delta^1$  or  $\delta^2$  the **computational path** of the run. Note that if  $M$  runs in polynomial time  $T$ , there are at most  $2^{T(|x|)}$  possible computational paths; in particular, the number of computational paths is finite.

The most fundamental complexity class for randomized computation is called BPP which stands for “bounded-error probabilistic polynomial time”.

**Definition 2.8** (BPP)

The complexity class BPP consists of all languages  $L$  for which there exists a probabilistic polynomial-time algorithm  $M$  such that the following holds:

- If  $x \in L$ , then  $\Pr [M(x) = 1] \geq \frac{2}{3}$ .
- If  $x \notin L$ , then  $\Pr [M(x) = 1] \leq \frac{1}{3}$ .

In both cases, the probabilities are taken with respect to the uniform distribution on the finite set of computational paths if  $M$  is applied to  $x$ .

### 2.1.4 Approximation Algorithms

Faced with the prospect that  $P \neq NP$  and its consequences for efficient solvability of practically highly relevant optimization problems such as MAX-CUT or TSP, one might be inclined to lower expectations in terms of polynomial-time solvability. The approach taken in this thesis is the one studied in the research area called *approximation algorithms* (see [22] for an extensive intro-

duction to the field): instead of looking for the actual solution of the combinatorial optimization problem, we are already satisfied with a guess obtained in polynomial time if it is guaranteed to satisfy a certain quality compared to the optimal solution.

**Definition 2.9** (Approximation Algorithms and Approximation Ratios)

Let  $\mathcal{P} = (\mathcal{I}, \mathcal{F}, \mathcal{S}, c)$  be a combinatorial optimization problem.

(i) An **approximation algorithm** for  $\mathcal{P}$  is a randomized polynomial-time algorithm that, given a problem instance  $I \in \mathcal{I}$ , outputs  $z \in \mathcal{F}(I)$ .

(ii) Let  $\mathcal{A}$  be an approximation algorithm for  $\mathcal{P}$  and  $I \in \mathcal{I}$ . Then if  $\max_{z \in \mathcal{F}(I)} c_I(z) \neq \min_{z \in \mathcal{F}(I)} c_I(z)$ , the **approximation ratio** of  $\mathcal{A}$  for  $I$  is defined as

$$\alpha_I(\mathcal{A}) = \frac{\mathbb{E}[c_I(\mathcal{A}(I))] - \min_{z \in \mathcal{F}(I)} c_I(z)}{\max_{z \in \mathcal{F}(I)} c_I(z) - \min_{z \in \mathcal{F}(I)} c_I(z)} \quad (2.2)$$

and as  $\alpha_I(\mathcal{A}) = 1$  if  $\max_{z \in \mathcal{F}(I)} c_I(z) = \min_{z \in \mathcal{F}(I)} c_I(z)$ .

(iii) Let  $\mathcal{A}$  be an approximation algorithm for  $\mathcal{P}$  and  $\mathcal{J} \subset \mathcal{I}$ . Then the **worst-case approximation ratio** of  $\mathcal{A}$  on  $\mathcal{J}$  is defined as

$$\alpha_{\mathcal{J}}(\mathcal{A}) = \inf_{I \in \mathcal{J}} \alpha_I(\mathcal{A}) \quad .$$

If  $\mathcal{J} = \mathcal{I}$ , we write  $\alpha(\mathcal{A})$  instead of  $\alpha_{\mathcal{I}}(\mathcal{A})$  and call this quantity the **worst-case approximation ratio** of  $\mathcal{A}$ .

It is obvious from this definition that  $\alpha_I(\mathcal{A}) \in [0, 1]$  for all  $I \in \mathcal{I}$ , with  $\alpha_I(\mathcal{A}) = 1$  corresponding to the case of  $\mathcal{A}$  outputting an exact solution; the goal is to find approximation algorithms that provide approximation ratios close to 1.

In the following, we will discuss some classical approximation algorithms. Most relevant to this thesis will be the MAX- $k$ -CUT problems, so we will put emphasis on the corresponding algorithms; a short discussion of approximation algorithms for other combinatorial optimization problems not extensively studied in this thesis follows to give an impression of the flavour and richness of the field.

### Classical approximation algorithms for the MAX-CUT problem

In view of (2.2), note that for both the unweighted MAX-CUT and MAX- $k$ -CUT problem, we have  $\min_{x \in \mathcal{F}(G)} c_G(x) = 0$  for any graph  $G$  since we can always assign the same value to all vertices which leads to none of the edges being satisfied.

- For the unweighted MAX-CUT problem applied to graphs  $G = (V, E)$ , there are two classical, randomized approximation algorithms of general interest.

The first algorithm is called the **trivial approximation algorithm**. It proceeds by randomly assigning values 0 and 1 to the vertices of  $G$ . We denote this algorithm by  $\mathcal{T}$ . It produces an approximation ratio of

$$\alpha_G(\mathcal{T}) \geq \frac{1}{2} \quad .$$

This can be easily seen by combining two simple observations:

- every edge is satisfied with probability  $1/2$  independently from any other edges
- the MAX-CUT value of a graph is trivially bounded from above by the number of edges, i.e.,  $\max_{z \in \mathcal{F}(G)} c_G(z) \leq |E|$ .

Combining these two, we obtain

$$\alpha_G(\mathcal{T}) = \frac{\mathbb{E}[c_G(\mathcal{T}(G))]}{\max_{z \in \mathcal{F}(G)} c_G(z)} = \frac{1/2 \cdot |E|}{\max_{z \in \mathcal{F}(G)} c_G(z)} \geq \frac{1/2 \cdot |E|}{|E|} = \frac{1}{2} \quad .$$

The second algorithm of particular interest is the **Goemans-Williamson algorithm** [17], from now on often abbreviated by “GW”. Given a graph  $G = ([n], E)$  with  $n \in \mathbb{N}$ , this randomized algorithm combines **semidefinite programming** with **randomized rounding** and proceeds in three steps:

1. Relax the combinatorial optimization problem to a semidefinite program (SDP) which is equivalent to the vector program

$$\max_{\{v_j\}_{j=1}^n \in \mathbb{S}^{n-1}} \frac{1}{2} \sum_{\{j,k\} \in E} (1 - \langle v_j, v_k \rangle) \quad ,$$

where  $\langle \cdot, \cdot \rangle$  denotes the inner product on  $\mathbb{R}^n$  (i.e.,  $\langle x, y \rangle := \sum_{k=1}^n x_k y_k$  for  $x, y \in \mathbb{R}^n$ ) and  $\mathbb{S}^{n-1} := \{v \in \mathbb{R}^n : \|v\|_2 := \sqrt{\langle v, v \rangle} = 1\}$ . Observe that the optimization is taken over *real-valued* unit vectors.

2. Solve this SDP using one of the polynomial-time methods available such as interior methods [23] or the ellipsoid algorithm [24] to obtain a solution  $v = \{v_j\}_{j=1}^n \in \mathbb{S}^{n-1}$ .
3. Sample  $r \in \mathbb{S}^{n-1}$  with respect to the **Haar measure** on  $\mathbb{S}^{n-1}$  (see [25] for a definition) and output the cut  $c: [n] \rightarrow \{0, 1\}$  via

$$c(j) = \begin{cases} 0 & \text{if } \langle v_j, r \rangle \geq 0, \\ 1 & \text{else.} \end{cases}$$

The main result of [17] is to establish the lower bound

$$\alpha(\text{GW}) \geq \frac{2}{\pi} \min_{0 \leq \theta \leq \pi} \frac{\theta}{1 - \cos \theta} \approx 0.8785 \quad .$$

Thus, the Goemans-Williamson algorithm significantly outperforms the trivial method. Although this algorithm is probabilistic due to the sampling in the third step, it can be derandomized, yielding a deterministic algorithm [26].

- However, better approximation ratios are achievable if we restrict our attention to certain subsets of graphs, an assumption often well justified in practice. Given a graph  $G = (V, E)$  and a vertex  $v \in V$ , we define the **degree** of  $v$  as the number of edges in  $G$  which contain  $v$ , and we will denote this quantity by  $\deg(v)$ . We call a graph  **$d$ -regular** if  $\deg(v) = d$  holds for all  $v \in V$ , and we denote the family of all  $d$ -regular graphs by  $\mathcal{D}_d$ .

As we will focus on (unweighted) 3-regular graphs in the Core Article II [2], we will briefly mention some results concerning  $\mathcal{D}_3$ . Using a simple greedy post-processing technique, Feige, Karpinski, and Langberg [15] exhibited an SDP-based approximation algorithm (in the following abbreviated by FKL) which provably achieves  $\alpha_{\mathcal{D}_3}(\text{FKL}) \geq 0.924$ . Shortly after, a refinement of this post-processing technique by Halperin, Livnat, and Zwick [16] yielded an improved approximation algorithm (in the following abbreviated by HLZ); they showed that  $\alpha_{\mathcal{D}_3}(\text{HLZ}) \geq 0.9326$  which is the best currently known algorithm for 3-regular graphs.

### Approximation algorithms for the MAX- $k$ -CUT problem

- Similarly as in the case of unweighted MAX-CUT, the unweighted MAX- $k$ -CUT problem also has a **trivial approximation algorithm** with approximation ratio  $1 - \frac{1}{k}$ ; it proceeds by randomly assigning values from the set  $\mathbb{Z}_k$  to the vertices of a given graph.
- Frieze and Jerrum [27] gave an algorithm for MAX- $k$ -CUT using SDP relaxations similar to the one in GW which achieves an approximation ratio of  $1 - \frac{1}{k} + 2 \log k/k^2$  for sufficiently large  $k$ . For such values of  $k$ , the algorithm is optimal in terms of approximation ratio if one assumes the ‘‘Unique Games Conjecture’’ [28] and  $\mathbf{P} \neq \mathbf{NP}$ , meaning that in this case, there is no approximation algorithm with a worst-case approximation ratio that is strictly larger than this one.
- Let us now focus on the case  $k = 3$ . For this problem, three algorithms by (a) Klerk, Pasechnik, and Warners (denoted by  $\text{KPW}_3$ ) [29], (b) Goemans and Williamson (denoted by  $\text{GW}_3$ ) [30], and (c) Newman (denoted by  $\text{Newman}_3$ ) [14] achieve the same currently known best approximation ratio of

$$\alpha(\text{KPW}_3) = \alpha(\text{GW}_3) = \alpha(\text{Newman}_3) \geq 0.836008 \quad .$$

In terms of analysis, Newman’s algorithm is the simplest one of the three. Furthermore, it is generalizable to larger  $k$ , and it performs only slightly worse than the algorithm by Frieze and Jerrum for large  $k$  [14]. Therefore, we will use this algorithm in Core article I) [1], where we study approximation algorithms for the MAX-3-CUT-problem.

## A brief overview over classical approximation algorithms for other combinatorial optimization problems

While all NP-complete combinatorial optimization problems are reducible to each other, they exhibit striking differences in terms of their approximability. As we have seen for the MAX- $k$ -CUT problems, it was possible to design deterministic approximation algorithms with constant worst-case approximation ratios; such combinatorial optimization problems form the complexity class APX. However, some combinatorial optimization problems such as the 0-1 KNAPSACK problem exhibit even stronger approximability guarantees: for this problem, one can find a so-called **fully polynomial-time approximation scheme (FPTAS)** which takes two inputs, a problem instance and some parameter  $\varepsilon > 0$ , and outputs an approximate solution with approximation ratio  $1 - \varepsilon$  in time  $O\left(p\left(\frac{1}{\varepsilon}, |x|\right)\right)$ , where  $p$  is some polynomial, therefore allowing arbitrarily good solutions in polynomial time [19]. The collection of such optimization problems is the complexity class FPTAS; if one only requires the run time to be  $O(q(|x|))$  with arbitrary dependence on  $\varepsilon$ , where  $q$  is some polynomial, one obtains a **polynomial-time approximation scheme**, the collection of which forms the class PTAS. It follows immediately from the definitions that the inclusions  $\text{FPTAS} \subseteq \text{PTAS} \subseteq \text{APX}$  hold; furthermore, it is known that equalities hold if and only if  $\text{P} = \text{NP}$ .

For combinatorial optimization problems not known to be in FPTAS such as MAX-CUT, one might wonder how well one can approximate them. For such problems, the **PCP theorem** [31,32] yields techniques to establish upper bounds on the approximability, provided that  $\text{P} \neq \text{NP}$  holds. For example, for the MAX-CUT problem, it is known that it is impossible to find a polynomial-time algorithm with a worst-case approximation ratio exceeding  $\frac{16}{17}$  unless  $\text{P} = \text{NP}$  [33], and if one additionally assumes the validity of the Unique Games Conjecture [28], it is known that the Goemans-Williamson algorithm is, in fact, optimal [34, 35], meaning that in that case, it is impossible to find a polynomial-time algorithm with a worst-case approximation ratio exceeding  $\alpha(\text{GW})$ . For more of such inapproximability results, we refer to [33]. Lastly, it is even possible to show that some problems such as TSP cannot be approximated within a constant unless  $\text{P} = \text{NP}$  [36].

## 2.2 Quantum Physics

Introduced in the 1920s as an attempt to answer fundamental physical questions that classical physics was unable to, quantum physics established itself over time as one of the most precise scientific disciplines. We will only state the very basics of quantum physics here; for a more in-depth introductory text on quantum physics, we refer to the literature, e.g., [37]. Before we dive into the foundations, we briefly introduce some Hilbert space terminology and notation.

### 2.2.1 Hilbert space essentials

Let  $\mathcal{H}$  be a complex vector space. We call  $\mathcal{H}$  a **complex inner product space** if it is equipped with an **inner product**  $\langle \cdot, \cdot \rangle: \mathcal{H} \times \mathcal{H} \rightarrow \mathbb{C}$  such that for all  $v_1, v_2, v_3 \in \mathcal{H}$  and  $\lambda \in \mathbb{C}$ , we have (i)  $\langle v_1, v_1 \rangle \geq 0$ , (ii)  $\langle v_1, v_2 \rangle = \overline{\langle v_2, v_1 \rangle}$ , and (iii)  $\langle v_1, v_2 + \lambda v_3 \rangle = \langle v_1, v_2 \rangle + \lambda \langle v_1, v_3 \rangle$ . If

$\mathcal{H}$  is additionally complete with respect to the induced norm  $\|\cdot\| := \sqrt{\langle \cdot, \cdot \rangle}$ , (i.e., all Cauchy sequences in  $\mathcal{H}$  converge with respect to the metric defined by  $\|\cdot\|$ ), we call it a **Hilbert space**. In the following, let  $\mathcal{H}$  be a Hilbert space. A linear function  $F: \mathcal{H} \rightarrow \mathbb{C}$  is called a **functional** on  $\mathcal{H}$ , and the set of bounded linear functionals on  $\mathcal{H}$ , i.e.,

$$\mathcal{H}^* := \{F | F: \mathcal{H} \rightarrow \mathbb{C} \text{ linear and there exists } C > 0 \text{ s. t. } |Fx| \leq C \cdot \|x\| \text{ for all } x \in \mathcal{H}\} \quad ,$$

is called the **dual space** of  $\mathcal{H}$ . The Riesz representation theorem [38] states that  $\mathcal{H}$  and  $\mathcal{H}^*$  are isometrically anti-isomorphic, implying that each functional  $F \in \mathcal{H}^*$  can be written uniquely as  $F(x) = \langle y, x \rangle$ , where  $y \in \mathcal{H}$ , so we can identify  $F$  with  $y$ . From now on, all Hilbert spaces in this thesis will be assumed to be finite-dimensional and complex unless specified otherwise.

In the following, we will need the concept of **tensor products of Hilbert spaces** to describe composite quantum systems. Given two (finite-dimensional) Hilbert spaces  $\mathcal{H}_1, \mathcal{H}_2$  with associated inner products  $\langle \cdot, \cdot \rangle_1, \langle \cdot, \cdot \rangle_2$  and two respective orthonormal bases  $\mathcal{B}_1 = \{e_j\}_{j=1}^{\dim \mathcal{H}_1}$ ,  $\mathcal{B}_2 = \{f_j\}_{j=1}^{\dim \mathcal{H}_2}$  (i.e.,  $\langle e_k, e_\ell \rangle = \delta_{k\ell}$  and  $\langle f_m, f_n \rangle = \delta_{mn}$  for all  $e_k, e_\ell \in \mathcal{B}_1, f_m, f_n \in \mathcal{B}_2$ ), we define the tensor product of  $\mathcal{H}_1$  and  $\mathcal{H}_2$  as the Hilbert space  $\mathcal{H}_1 \otimes \mathcal{H}_2$  that is characterized by the orthonormal basis

$$\{e_j \otimes f_k\}_{\substack{j \in [\dim \mathcal{H}_1], \\ k \in [\dim \mathcal{H}_2]}}$$

and the inner product

$$\langle \phi_1 \otimes \phi_2, \psi_1 \otimes \psi_2 \rangle = \langle \phi_1, \psi_1 \rangle_1 \cdot \langle \phi_2, \psi_2 \rangle_2 \quad ,$$

where  $\phi_1, \psi_1 \in \mathcal{B}_1, \phi_2, \psi_2 \in \mathcal{B}_2$ . The inner product of two general elements of  $\mathcal{H}_1 \otimes \mathcal{H}_2$  is then fully determined due to linearity; it can be shown that this definition of the inner product does not depend on the particular choice of the orthonormal basis. Note that the symbol  $\otimes$  denotes both the tensor product of vector spaces and vectors. Concretely, the general **tensor product of two vectors** (also known as the **Kronecker product**) is defined as follows: given  $x = \sum_{j=1}^{\dim \mathcal{H}_1} x_j e_j \in \mathcal{H}_1$  and  $y = \sum_{k=1}^{\dim \mathcal{H}_2} y_k f_k \in \mathcal{H}_2$ , where  $\{x_j\}_{j \in [\dim \mathcal{H}_1]}, \{y_k\}_{k \in [\dim \mathcal{H}_2]} \subset \mathbb{C}$ , we define

$$x \otimes y := \sum_{j=1}^{\dim \mathcal{H}_1} \sum_{k=1}^{\dim \mathcal{H}_2} (x_j y_k) e_j \otimes f_k \quad .$$

It is straightforward to extend the definitions to tensor products of more than two Hilbert spaces. The definition of the tensor product implies directly that  $\dim \bigotimes_{k=1}^n \mathcal{H}_k = \prod_{k=1}^n \dim \mathcal{H}_k$  for  $n$  Hilbert spaces  $\mathcal{H}_1, \dots, \mathcal{H}_n$ ; in particular, the dimension grows exponentially with the number of Hilbert spaces.

### 2.2.2 Operator terminology

For a Hilbert space  $\mathcal{H}$  with inner product  $\langle \cdot, \cdot \rangle$ , a linear function  $A: \mathcal{H} \rightarrow \mathcal{H}$  is called an **operator** on  $\mathcal{H}$ , and the **space of bounded linear operators** on  $\mathcal{H}$  is defined as

$$\mathcal{B}(\mathcal{H}) := \{A | A: \mathcal{H} \rightarrow \mathcal{H} \text{ linear and there exists } C > 0 \text{ s. t. } \|Ax\| \leq C \cdot \|x\| \text{ for all } x \in \mathcal{H}\}.$$

We denote the identity operator on  $\mathcal{H}$  by  $I$  or, if context requires it, by  $I_{\mathcal{H}}$ .

Let  $A \in \mathcal{B}(\mathcal{H})$  and  $M$  be the matrix representation of  $A$  with respect to some orthonormal basis  $\mathcal{O}$  of  $\mathcal{H}$ . The **trace** of  $A$  is defined as  $\text{tr}[A] = \sum_k M_{kk}$ , i.e., the sum of the elements on the diagonal of  $M$ . The **adjoint** of  $A$  is defined as the unique linear operator  $A^\dagger \in \mathcal{B}(\mathcal{H})$  for which  $\langle A^\dagger x, y \rangle = \langle x, Ay \rangle$  holds for all  $x, y \in \mathcal{H}$ . We call  $A$  **Hermitian** or an **observable** if  $A^\dagger = A$ , **unitary** if  $A^\dagger A = I$ , **positive semidefinite** if  $\langle x, Ax \rangle \geq 0$  for all  $x \in \mathcal{H}$ , or a **projector** if  $A^2 = A$ . The dimension of the image of  $A$  is called the **rank** of  $A$ . We denote the set of all unitary operators by  $U(\mathcal{H})$ ; it forms a group under composition. For two operators  $A, B \in \mathcal{B}(\mathcal{H})$ , we write that  $A \leq B$  if  $B - A$  is positive semidefinite. Note that  $A^\dagger$  is represented by the matrix  $M^\dagger$  with respect to the basis  $\mathcal{O}$ , where  $M_{jk}^\dagger = \overline{M_{kj}}$ . This also extends the definition of an adjoint to linear maps which have different finite-dimensional domains and codomains.

Now let  $\mathcal{H}_1, \mathcal{H}_2$  be two (finite-dimensional) Hilbert spaces and  $A \in \mathcal{B}(\mathcal{H}_1), B \in \mathcal{B}(\mathcal{H}_2)$ . Then we define the **operator tensor product** of  $A$  and  $B$  as the operator  $A \otimes B \in \mathcal{B}(\mathcal{H}_1 \otimes \mathcal{H}_2)$  that acts as

$$(A \otimes B)(x \otimes y) = Ax \otimes By$$

for any  $x \in \mathcal{H}_1, y \in \mathcal{H}_2$ . Due to linearity, this definition fully determines the action for general elements of  $\mathcal{H}_1 \otimes \mathcal{H}_2$ . The definition of tensor products of operators extends in the natural way to more than two operators, and a notational convention for such tensor products is to omit the symbols  $\otimes$  and identities and use indices to indicate the label of the Hilbert space that the operator is acting on nontrivially; for example, instead of writing  $A \otimes B \otimes I \otimes C \otimes I$ , one writes  $A_1 B_2 C_4$ , where the fact that the operator is acting on a tensor product of five Hilbert spaces has to be made clear in context. We define the **support** of an operator  $O$  (denoted by  $\text{supp}(O)$ ) to be the set of labels of Hilbert spaces on which  $O$  is acting on nontrivially; for example,  $\text{supp}(A_1 B_2 C_4) = \{1, 2, 4\}$ .

We will also need the notion of complete positivity. A map  $A \in \mathcal{B}(\mathcal{B}(\mathcal{H}_1))$  is called **completely positive** if  $A \otimes I_{\mathcal{B}(\mathcal{H}_2)}$  is positive semidefinite for any finite-dimensional Hilbert space  $\mathcal{H}_2$ .

### 2.2.3 Dirac notation

The quantum physics community makes extensive use of the so-called **Dirac** or **bra-ket notation** instead of the mathematical notation used so far. To introduce it, let  $\mathcal{H}$  be a complex Hilbert space with inner product  $\langle \cdot, \cdot \rangle$  and  $u, v, w \in \mathcal{H}$ . In Dirac notation, we write  $|u\rangle$  and  $|v\rangle$  instead of  $u$  and  $v$ ; furthermore, as mentioned in Subsection 2.2.1,  $w$  defines an element of  $\mathcal{H}^*$  denoted by  $\langle w|$  via the Riesz isomorphism, i.e.,  $(\langle w|)(v) = \langle w, v \rangle$ . The expression  $\langle w|$  is called a **bra** and the expression  $|u\rangle$  is called a **ket**. Furthermore, we denote the inner product of  $|u\rangle$

and  $|v\rangle$  as  $\langle u|v\rangle$  instead of  $\langle u, v\rangle$ , and we denote the **outer product** of  $|v\rangle$  and  $\langle w|$  by  $|v\rangle\langle w|$  and define it as the operator on  $\mathcal{H}$  acting as  $|v\rangle\langle w|(|u\rangle) := \langle w|u\rangle \cdot |v\rangle$ . Concerning tensor products, we will often write either  $|u\rangle \otimes |v\rangle$ ,  $|u\rangle|v\rangle$ , or  $|uv\rangle$  to denote the tensor product of  $|u\rangle$  and  $|v\rangle$  in Dirac notation, and if necessary, we will use subscripts to indicate the label of the associated Hilbert space (e.g.,  $|u\rangle_1 \otimes |v\rangle_2$ ,  $|u, v\rangle_{1,2}$ ,  $|uv\rangle_{1,2}$ , or  $|u\rangle_1|v\rangle_2$ ).

We will mostly work with the Hilbert space  $\mathcal{H} = \mathbb{C}^d$ , where  $d \in \mathbb{N}$ ; we then say that  $\mathcal{H}$  models a **qudit** of level  $d$ , or a **qubit** if  $d = 2$ . For this Hilbert space, the canonical basis is also called the **computational basis**, i.e., it is given by  $\{|j\rangle\}_{j \in \mathbb{Z}_d}$ , where for  $k \in \mathbb{Z}_d$ , the  $(k+1)$ -st entry of  $|j\rangle$  is given by  $|j\rangle_{k+1} = \delta_{jk}$ . This notion extends to the tensor product case  $\mathcal{H} = (\mathbb{C}^d)^{\otimes n}$  with  $n \in \mathbb{N}$ : the computational basis is then given by  $\{|j\rangle\}_{j \in \mathbb{Z}_d^n}$ , i.e., each basis element is labeled by a  $d$ -ary string of length  $n$ . For example,  $|0\rangle \in \mathbb{C}^2$ ,  $|1\rangle \in \mathbb{C}^2$ , and  $|2001\rangle \in (\mathbb{C}^3)^{\otimes 4}$  represent in conventional notation the vectors

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad \text{and} \quad |2001\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}.$$

#### 2.2.4 The postulates of quantum physics

In general, **quantum systems** are described by complex Hilbert spaces, so-called **state spaces**. In this thesis, all state spaces will be of finite dimension, and all quantum systems will be **closed**, i.e., they fully describe the physical experiment or computation that we are interested in and we, the experimenter, have full control over the system. A **quantum state** is described by a linear, positive semidefinite operator  $\rho$  with  $\text{tr}[\rho] = 1$ , where  $\rho$  acts on the state space  $\mathcal{H}$ ; such operators are called **density operators**. If  $\rho$  is a rank-1 projector onto a space spanned by a vector  $|\psi\rangle \in \mathcal{H}$  with  $\langle \psi|\psi\rangle = 1$ , we call  $\rho = |\psi\rangle\langle \psi|$  a **pure state** and will instead work directly with  $|\psi\rangle$ ; if  $\rho$  is not pure, we will call it a **mixed state**. Note that there is some ambiguity for pure states  $|\psi\rangle$  as all vectors of the form  $e^{i\varphi}|\psi\rangle$  with  $\varphi \in \mathbb{R}$  will give rise to the same density operator and are therefore physically indistinguishable; for this reason, we identify all such states with each other and call the respective multiplicative prefactor a **global phase**. One can interpret mixed states as **ensembles of pure states** in the following way: the spectral theorem [38] implies that we can write  $\rho = \sum_{k=1}^{\dim \mathcal{H}} \lambda_k |\psi_k\rangle\langle \psi_k|$ , where  $\lambda_1, \dots, \lambda_{\dim \mathcal{H}}$  are the eigenvalues of  $\rho$  with associated orthonormal eigenvectors  $\{|\psi_k\rangle\}_{k=1}^{\dim \mathcal{H}} \subset \mathcal{H}$ . Since  $\rho$  is positive semidefinite, we know that  $\{\lambda_k\}_{k=1}^{\dim \mathcal{H}} \in \mathbb{R}_{\geq 0}$ , and since  $\text{tr}[\rho] = 1$ , we know that  $\sum_{k=1}^{\dim \mathcal{H}} \lambda_k = 1$ . Operationally, we can interpret this as preparing the quantum system in the pure state  $|\psi_k\rangle$  with probability  $\lambda_k$ .

To model the physical evolution of a quantum system, we will exclusively work within the quantum channel framework. To this end, let  $\mathcal{H}_1$  and  $\mathcal{H}_2$  be two state spaces. A **quantum channel** is a completely positive map  $\mathcal{E}: \mathcal{B}(\mathcal{H}_1) \rightarrow \mathcal{B}(\mathcal{H}_2)$  with  $\text{tr}[\mathcal{E}(\rho)] = \text{tr}[\rho]$  for all  $\rho \in \mathcal{B}(\mathcal{H}_1)$ . It is well known [39] that every completely positive map  $\mathcal{E}$  can be described via **Kraus operators**  $\{E_k\}_k$ , where the  $E_k: \mathcal{H}_1 \rightarrow \mathcal{H}_2$  are linear maps which satisfy  $\sum_k E_k^\dagger E_k = I_{\mathcal{H}_1}$ ; we can



then express the associated action of  $\mathcal{E}$  as  $\mathcal{E}(\rho) = \sum_k E_k \rho E_k^\dagger$ . Of utmost importance in this thesis will be **unitary evolutions**, where the Kraus representation consists of a single unitary Kraus operator  $U \in \mathcal{U}(\mathcal{H})$ ; applied to a system in the mixed state  $\rho$ , the resulting state will therefore be  $U\rho U^\dagger$ , while for a pure state  $|\psi\rangle$ , the resulting state will be  $U|\psi\rangle$ . Note that for a unitary evolution, domain and codomain of the map must be the same Hilbert space  $\mathcal{H}$ . Another way to describe the unitary evolution of a closed quantum system is via the **Schrödinger equation** which is the initial value problem

$$i \frac{d}{dt} |\psi(t)\rangle = H(t) |\psi(t)\rangle \quad , \quad |\psi(t_0)\rangle = |\psi_0\rangle$$

where  $t_0 \in \mathbb{R}$  is the initial time of the system,  $T \in \mathbb{R}$  is the maximal evolution time,  $|\psi(t)\rangle$  is the state of the system at time  $t \in [t_0, T]$ ,  $|\psi_0\rangle$  is some given initial state of the system, and  $H: [t_0, T] \rightarrow \mathcal{B}(\mathcal{H})$  is a map with the property that  $H(t)$  is Hermitian for all  $t \in [t_0, T]$ . This map is called the **Hamiltonian** of the system. Solving this differential equation means to determine the unitary evolution operator  $U: [t_0, T] \rightarrow \mathcal{U}(\mathcal{H})$  such that  $|\psi(t)\rangle = U(t)|\psi_0\rangle$  for all  $t \in [t_0, T]$  and all  $|\psi_0\rangle \in \mathcal{H}$ . For a time-independent Hamiltonian  $H$ , it is easy to see that

$$U(t) = \exp(-iH(t - t_0))$$

is the corresponding unitary evolution operator, while for time-dependent Hamiltonians, finding this operator involves so-called time-ordered integrals [40].

Besides unitary evolutions, the most used quantum channels in this thesis are **measurements**. These are described by Kraus operators  $\{M_m\}_m$  satisfying  $\sum_m M_m^\dagger M_m = I$ , where the index of the measurement corresponds to the **outcome** of the measurement which can be read out by the experimenter. For mixed states, if the quantum system is in the state  $\rho$  and it is measured with respect to  $\{M_m\}_m$ , the measurement result  $m$  is obtained with probability  $\text{tr}[M_m^\dagger M_m \rho]$  and the resulting post-measurement state of the system is  $\frac{M_m \rho M_m^\dagger}{\text{tr}[M_m^\dagger M_m \rho]}$ . For pure states, this implies that if the quantum system is in the state  $|\psi\rangle$  and it is measured with respect to  $\{M_m\}_m$ , the measurement result  $m$  is obtained with probability  $\langle \psi | M_m^\dagger M_m | \psi \rangle$  and the post-measurement state of the system is  $\frac{M_m |\psi\rangle}{\sqrt{\langle \psi | M_m^\dagger M_m | \psi \rangle}}$ . We can define a set of measurement operators via an observable  $O \in \mathcal{B}(\mathcal{H})$  by writing  $O = \sum_{m \in E} \lambda_m M_m$ , where  $\{\lambda_m\}_{m \in E} \subset \mathbb{R}$  is the set of distinct eigenvalues of  $O$  with label set  $E$  and  $\{M_m\}_{m \in E}$  are the projectors onto the associated eigenspaces.

The last basic postulate of quantum physics describes how one can combine quantum systems to obtain a **composite quantum system**. This is done via the tensor product: for  $n$  **subsystems**  $\mathcal{H}_1, \dots, \mathcal{H}_n$ , the state space of the combined quantum system is given by  $\mathcal{H}_1 \otimes \dots \otimes \mathcal{H}_n$ .

### 2.2.5 Further remarks

1. While it is obvious that for two states  $|\psi_1\rangle \in \mathcal{H}_1$  and  $|\psi_2\rangle \in \mathcal{H}_2$ , the tensor product  $|\psi_1\rangle \otimes |\psi_2\rangle$  is in  $\mathcal{H} \otimes \mathcal{H}$ , not every element of the tensor product space is of this form. In

general, we call  $|\psi\rangle \in \bigotimes_{j=1}^n \mathcal{H}_j$  a **product state** if  $|\psi\rangle$  can be written as  $|\psi\rangle = \bigotimes_{j=1}^n |\psi_j\rangle$ , where  $|\psi_j\rangle \in \mathcal{H}_j$  for all  $j \in [n]$ . If it cannot be written in this way, we call  $|\psi\rangle$  **entangled** with respect to this decomposition. By a counting argument, it is easy to see that most states are, in fact, entangled. Probably the most famous example for an entangled state is the **EPR** pair  $|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ .

2. In this thesis, all state spaces will be of the form  $\mathcal{H} = (\mathbb{C}^d)^{\otimes n}$  for some  $d, n \in \mathbb{N}$ , and if  $d = 2$ , all measurements will be with respect to the computational basis, i.e., we measure each qudit with respect to the observable  $Z$  defined as

$$Z = |0\rangle\langle 0| - |1\rangle\langle 1| = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix},$$

where the matrix representation on the right hand side is with respect to the computational basis. The eigenvalues of  $Z$  are  $\lambda_0 = 1$  and  $\lambda_1 = -1$ , and we denote the possible measurement outcomes by 0 and 1.

3. Another commonly used observable for single-qubit measurements is the operator  $X$ , where – in the qubit case –  $X$  can be expressed as the matrix

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

with respect to the computational basis. After such a measurement with outcome  $m \in \{+, -\}$ , the system after the measurement is in the state  $|m\rangle$ , where

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad \text{and} \quad |-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

are the +1 and –1 eigenstates of  $X$ .

4. In the general qudit case with  $d \in \mathbb{N}$ ,  $X$  and  $Z$  can be expressed as

$$X = \sum_{k \in \mathbb{Z}_d} |k+1\rangle\langle k| \quad \text{and} \quad Z = \sum_{k \in \mathbb{Z}_d} e^{2\pi ki/d} |k\rangle\langle k| \quad ;$$

however, they are Hermitian if and only if  $d = 2$ . These operators together with  $Y = iXZ$  are called **Pauli operators**, and together with the identity  $I_{\mathbb{C}^d}$ , they form a basis of  $\mathcal{B}(\mathbb{C}^d)$ .

## 2.3 Quantum Computation

Finally, we discuss the main topic of interest in this thesis, namely quantum computation. For a more in-depth introduction, we recommend the standard textbook by Chuang and Nielsen [39], but other texts are also available such as [41].

### 2.3.1 Quantum Circuits

While it is possible to define quantum versions of Turing machines [5], we will instead work exclusively with the **quantum circuit model**. We will also restrict our treatment to quantum circuits which implement unitary operations, i.e., there will be no measurements except at the very end of the circuit; the “principle of deferred measurement” [39] allows us to do so without loss of generality. Furthermore, in this section, we will work exclusively with pure states and focus on the **qubit model** (i.e., the state space will be  $\mathcal{H} = (\mathbb{C}^2)^{\otimes m}$  for  $m \in \mathbb{N}$ ), but the generalization to mixed states and/or qudits of higher dimension is rather straightforward.

In general, a quantum algorithm is implemented by first applying a unitary operator  $U \in \mathcal{U}\left((\mathbb{C}^2)^{\otimes m}\right)$  to a (pure) input state  $|\psi\rangle \in (\mathbb{C}^2)^{\otimes m}$ , where  $m$  is the input size, and then measuring the resulting state  $U|\psi\rangle$  with respect to an observable; this yields a classical output string. Note that the first of these steps (i.e., the application of  $U$  to  $|\psi\rangle$ ) is sometimes also called **state preparation**. However, a direct application of  $U$  is often not possible; in practice, one only has access to a specific set (called **gate set**) of unitary operators (called **gates**) that one can apply to the system, often with each gate acting nontrivially only on a small set of qubits. A popular example for a gate set on  $m$  qubits is given by

$$\mathcal{G}_m = \{\text{CNOT}_{j,k}\}_{\substack{j,k=1 \\ j \neq k}}^m \cup \{H_j, T_j, S_j\}_{j=1}^m \quad , \quad (2.3)$$

where the subscripts denote the label(s) of the qubit(s) that the gate is acting on nontrivially, and the actions of the operators on computational basis states  $|x\rangle_j, |y\rangle_k \in \{|0\rangle, |1\rangle\}$ , where  $j \neq k \in [m]$ , are defined as follows:

$$\begin{aligned} \text{CNOT}_{j,k}|x\rangle_j|y\rangle_k &= |x\rangle_j|x \oplus y\rangle_k \quad , \\ H_j|x\rangle_j &= \frac{1}{\sqrt{2}}(|0\rangle_j + (-1)^x|1\rangle_j) \quad , \\ S_j|x\rangle_j &= i^x|x\rangle_j \quad , \\ T_j|x\rangle_j &= e^{i\frac{\pi}{4}x}|x\rangle_j \quad . \end{aligned}$$

Concerning terminology, we call the first qubit in the definition of the CNOT gate the **control qubit** and the second qubit the **target qubit**. This terminology is borrowed from classical computation since the value of the first bit determines the output of the second bit; however, note that this classical behaviour does not necessarily transfer to the quantum setting if one works in bases other than the computational basis [39].

The popularity of this gate set originates in the fact [39] that it is an **universal gate set**, i.e.,  $\mathcal{G}_m$  is a finite subset of  $\mathcal{U}\left((\mathbb{C}^2)^{\otimes m}\right)$  such that for every  $\varepsilon > 0$  and every  $U \in \mathcal{U}\left((\mathbb{C}^2)^{\otimes m}\right)$ , there exist  $k \in \mathbb{N}$  and a finite sequence  $g_1, \dots, g_k \in \mathcal{G}_m$  such that  $\|U - g_1 \dots g_k\| < \varepsilon$ , where  $\|\cdot\|$  denotes the operator norm defined as  $\|A\| := \sup_{x \in \mathcal{H}} \frac{\|Ax\|}{\|x\|}$  for a linear operator  $A$  acting on a Hilbert space  $\mathcal{H}$ .

Let us assume that we are given some gate set  $\mathcal{G}_m$  and that we can write the unitary operator that we want to implement as  $U = g_1 \dots g_k$ , where  $k \in \mathbb{N}$  and  $g_1, \dots, g_k \in \mathcal{G}_m$ . To define quantum

circuits in the qubit models, we will use **directed graphs**. Recall that a directed graph is a graph  $G = (V, E)$  in which  $E \subseteq V \times V$ , i.e., each edge  $e = (u, v)$  is equipped with a direction and pointing from vertex  $u$  to vertex  $v$ . The **in-degree** of a vertex  $u$  is the number of edges that end in  $u$  while the **out-degree** of  $u$  is the number of edges that start in  $u$ ; these quantities are denoted by  $\text{indeg}(u)$  and  $\text{outdeg}(u)$ , respectively. A vertex is called a **source** if its in-degree is 0 and a **sink** if its out-degree is 0. Furthermore, we call a  $k$ -tuple  $p = (p_1, \dots, p_k) \in E^k$  a **directed path** of length  $k$  if for any  $\ell \in [k-1]$ , the second entry of  $p_\ell$  is equal to the first entry of  $p_{\ell+1}$ ; if additionally the second entry of  $p_k$  is equal to the first entry of  $p_1$ , we call  $p$  a **cycle** of length  $k$ . We call  $G$  **acyclic** if it does not contain any cycles.

With these definitions, we can now define quantum circuits in the qubit model as follows:

**Definition 2.10** (Quantum Circuits)

Let  $m \in \mathbb{N}$  and  $\mathcal{G}_m$  be a gate set on  $m$  qubits. We call a directed, acyclic graph an  *$m$ -qubit quantum circuit* if it satisfies the following conditions:

- (i) The graph contains  $m$  sources  $u_1, \dots, u_m$  with  $\text{outdeg}(u_k) = 1$  for all  $k \in [m]$ , and each  $u_k$  is labeled with  $k$ .
- (ii) The graph contains  $m$  sinks  $v_1, \dots, v_m$  with  $\text{indeg}(v_k) = 1$  for all  $k \in [m]$ , and each  $v_k$  is labeled with a single-qubit observable.
- (iii) Each nonsource, nonsink vertex  $w$  satisfies  $\text{indeg}(w) = \text{outdeg}(w)$  and is labeled with an  $\text{indeg}(w)$ -qubit gate  $g \in \mathcal{G}_m$ .

To use a quantum circuit to perform computation on a pure input state  $|\psi\rangle \in (\mathbb{C}^2)^{\otimes m}$ , one follows the edges of the graph starting in the sources and performs at each nonsource, nonsink vertex the unitary action specified by the label to the appropriate qubits. At the end of the circuit, the single-qubit measurements at the sinks are performed, yielding the result of the computation.

Figure 2.1(a) shows an example for the directed, acyclic graph associated with a quantum circuit; however, in practice, one rather uses so-called **quantum circuit diagrams** to visualize the circuits rather than graphs. An example for such a diagram is given in Figure 2.1(b).

Just as in the classical case, we want to quantify the performance of a quantum algorithm in terms of the resources it uses. In the quantum circuit model, there are two main quantities of interest to us. Given two directed paths  $p, q$  in  $G$ , we say that  $p$  is longer than  $q$  if the length of  $p$  is greater than the length of  $q$ .

**Definition 2.11** (Circuit Size and Circuit Depth)

Let  $C$  be a quantum circuit and  $G$  be the directed, acyclic graph describing  $C$ .

- (i) We call the number of nonsource, nonsink vertices in  $G$  (i.e., the number of gates in  $C$ ) the **size** of  $C$  and denote it by  $|C|$ .
- (ii) We call the number of gates on a longest directed path from a source to a sink the **depth** of  $C$  and denote it by  $d(C)$ .

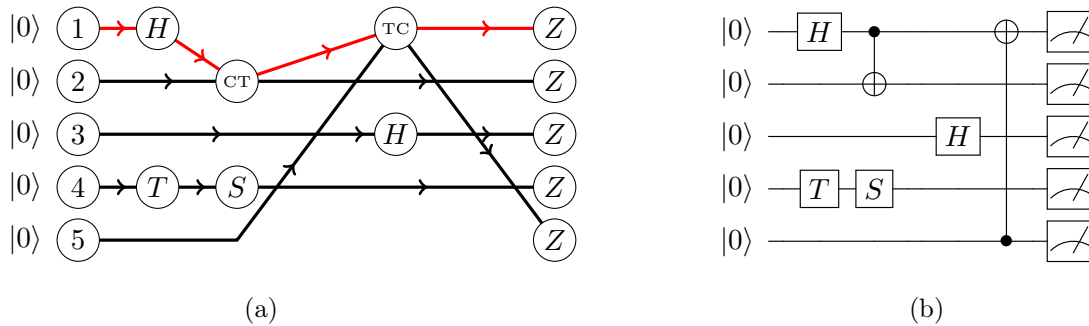


Figure 2.1: An example for a quantum circuit acting on  $m = 5$  qubits using the gate set  $\mathcal{G}_5$  defined in (2.3). (a) The directed, acyclic graph associated with the circuit, where the vertices are indicated by circles. The sources are situated on the left side of the graph and labeled with the numbers of the associated qubits; they are followed by gates which are labeled by the respective gate names, where CT denotes the CNOT gate such that the label of the control qubit is smaller than the one of the target qubit and TC denotes the CNOT gate in which those roles are reversed. The sinks are situated on the right side of the graph; each of them is labeled with the single-qubit observable  $Z$ . We indicate an application of the circuit to the product input state  $|0\rangle^{\otimes 5}$  by writing  $|0\rangle$  to the left of each associated source. A longest directed path in the graph is indicated in red. (b) The associated circuit diagram of the circuit. CNOT gates are indicated by the symbols connecting a dot (the control) and a  $\oplus$  symbol (the target). The meter symbols at the end of the circuit indicate a measurement with respect to the computational basis.

In the circuit  $C$  shown in Figure 2.1, we have  $|C| = 6$  and  $d(C) = 3$ , with a longest directed path from a source to a sink indicated in red.

### 2.3.2 Quantum Algorithms

Let  $\mathcal{P} = (\mathcal{I}, \mathcal{F} = \{\mathcal{F}(I)\}_{I \in \mathcal{I}}, \mathcal{S} = \{\mathcal{S}(I)\}_{I \in \mathcal{I}})$  be a computational problem. To apply a quantum algorithm to a problem instance  $I \in \mathcal{I}$  of size  $n$ , one first encodes  $I$  into an appropriately chosen quantum circuit on  $m$  qubits, where  $m \geq n$ , and then applies this quantum circuit to an appropriately chosen input state  $|\psi\rangle \in (\mathbb{C}^2)^{\otimes m}$  to obtain a classical output  $\{0, 1\}^m$ .

In this description, we did not require that  $m = n$ . The reason for this is that one might be interested in performing auxiliary computations as part of the circuit which might require more than  $n$  qubits. These  $m - n$  qubits are called **auxiliary qubits**.

Furthermore, note that for inputs of different sizes, the dimensions of the state spaces used typically differ, so one has to use different quantum circuits for the computation. In contrast, the classical Turing machine model introduced in Section 2.1.2 is capable of performing an algorithm on any input size; it is therefore a **uniform** model of computation, while the general circuit model is a **nonuniform** model of computation. To describe an algorithm in the quantum circuit model, we therefore need to provide a quantum circuit for every input size  $n$ ; such a collection of quantum circuits is called a **circuit family** and is usually denoted by  $\mathcal{C} = \{\mathcal{C}_n\}_{n \in \mathbb{N}}$ .

There is another subtlety in the circuit model that is not present in uniform computational models: given an input size  $n$ , one first has to build the circuit before one can apply the quantum algorithm. This means that one could “cheat” by hiding most of the complexity of a computa-

tional problem in building the circuit, leaving a very low-depth or small quantum circuit. To circumvent this, one imposes a restriction on the procedure used to build the circuits.

**Definition 2.12** (Uniform Quantum Circuit Families)

We call a set  $\mathcal{C} = \{\mathcal{C}_n\}_{n \in \mathbb{N}}$  a **uniform quantum circuit family** if each  $\mathcal{C}_n$  is an  $m(n)$ -qubit quantum circuit (i.e.,  $m \in \mathbb{N}$  depends on  $n$ ), and there is a classical deterministic polynomial-time algorithm that computes the mapping  $n \mapsto \mathcal{C}_n$ .

We can now adapt Definitions 2.5 and 2.7 to the circuit setting.

**Definition 2.13** (Quantum Circuit Complexity and Depth)

Let  $\mathcal{C} = \{\mathcal{C}_n\}_{n \in \mathbb{N}}$  be uniform quantum circuit family and  $T, D: \mathbb{N} \rightarrow \mathbb{N}$ .

- (i) We say that  $\mathcal{C}$  **has circuit complexity**  $T$  if  $|\mathcal{C}_n| = T(n)$  for all  $n \in \mathbb{N}$ .
- (ii) We say that  $\mathcal{C}$  **has circuit depth**  $D$  if  $d(\mathcal{C}_n) = D(n)$  for all  $n \in \mathbb{N}$ .
- (iii) We say that  $\mathcal{C}$  is a **polynomial-time quantum algorithm** if there is a polynomial  $T$  such that  $\mathcal{C}$  has circuit complexity  $T$ .

As discussed in Section 2.2.4, quantum theory is probabilistic in its nature due to the measurement postulate. This suggests defining quantum complexity classes via probabilistic means, so it is more natural to adapt BPP to the quantum setting instead of P. We therefore define the following complexity class in the spirit of Definition 2.8.

**Definition 2.14** (BQP)

The complexity class **BQP** is the set of all languages  $L$  for which there exists a polynomial-time quantum algorithm specified by a uniform circuit family  $\mathcal{C} = \{\mathcal{C}_n\}_{n \in \mathbb{N}}$ , where each  $m(n)$ -qubit circuit  $\mathcal{C}_n$  is specified in terms of the gate set  $\mathcal{G}_{m(n)}$  defined in (2.3), such that the following holds:

- If  $x \in L$ , then  $\Pr[\mathcal{C}_n(x)_1 = 1] \geq \frac{2}{3}$ .
- If  $x \notin L$ , then  $\Pr[\mathcal{C}_n(x)_1 = 1] \leq \frac{1}{3}$ .

Here,  $\mathcal{C}_n(x)_1 \in \{0, 1\}$  denotes the measurement result of the first qubit obtained by applying the quantum circuit  $\mathcal{C}_n$  with input  $x \in \{0, 1\}^n$ .

It is evident that  $\text{BPP} \subseteq \text{BQP}$ , and it is conjectured that the inclusion is indeed proper with the decision problem associated with factoring being known to be in BQP due to Shor's algorithm [8] while no classical, probabilistic, efficient algorithm has been found for the problem. However, it is also conjectured that  $\text{NP} \not\subseteq \text{BQP}$ , so the power of quantum computation presumably does not allow solving NP-complete problems.

## Chapter 3

# The Quantum Approximate Optimization Algorithm (QAOA)

In this chapter, we give a detailed overview over the main quantum algorithm studied in this thesis: the Quantum Approximate Optimization Algorithm (abbreviated “QAOA”) introduced by Edward Farhi, Jeffrey Goldstone, and Sam Gutmann in [12].

### 3.1 Definition of QAOA

#### 3.1.1 The Quantum Adiabatic Algorithm (QAA)

The core underlying motivation for QAOA comes from **Quantum Adiabatic Computation** [42] which itself is derived from the concept of **adiabatic evolution**. The setting is as follows: we are given a Hamiltonian  $H_0$  on a finite-dimensional quantum system  $\mathcal{H}$  that is in the state  $|\psi_0\rangle$ , where  $|\psi_0\rangle$  is a **ground state** of  $H_0$ , i.e., it is an eigenstate of  $H_0$  with the property that its associated eigenvalue is the minimum of all eigenvalues of  $H_0$ . Now let  $T > 0$  and define the time-dependent Hamiltonian  $H_T: [0, T] \rightarrow \mathcal{B}(\mathcal{H})$  as

$$H_T(t) = (1 - t/T)H_0 + (t/T)H_1 \quad , \quad (3.1)$$

where  $H_1$  is another Hamiltonian on  $\mathcal{H}$ . As we obviously have  $H_T(0) = H_0$  and  $H_T(T) = H_1$ , we ask whether there are sufficient conditions such that the state  $|\psi_T\rangle$  of the system after unitary evolution governed by  $H_T$  is close to a ground state of  $H_1$ .

This question is answered by the **adiabatic theorem** [43]. To state it, let  $E_k(s)$  be the  $k$ -th smallest eigenvalue of  $H_T(s)$ , where  $k \in \{1, \dots, \dim(\mathcal{H})\}$  and  $s \in [0, T]$  (i.e.,  $E_1(s) \leq E_2(s) \leq \dots \leq E_{\dim(\mathcal{H})}(s)$ ). Now define the **minimum spectral gap**  $\Delta$  as

$$\Delta := \min_{s \in [0, T]} (E_2(s) - E_1(s)) \quad .$$

The adiabatic theorem now states that if  $\Delta > 0$ , then

$$\lim_{T \rightarrow \infty} |\langle \psi_0 | \psi_T \rangle| = 1 \quad .$$

Quantitatively, the proof of the result shows that  $T$  needs to grow quadratically with  $\Delta^{-1}$  to make  $|\langle \psi_0 | \psi_T \rangle|$  close to 1. The adiabatic theorem motivates the following quantum algorithmic approach to find solutions of combinatorial optimization problems: first, we choose  $H_0$  to be a Hamiltonian whose ground state  $|\psi_0\rangle$  is simple to prepare (e.g., a product state). In the following, this Hamiltonian will be called the **mixing Hamiltonian** and denoted by  $H_M$ . If we now manage to encode a solution of an instance  $I$  of a combinatorial optimization problem  $\mathcal{P}$  as the ground state of a Hamiltonian  $H_1$ , which we call the **problem Hamiltonian** and denote by  $H_I^{\mathcal{P}}$  from now on, we can adiabatically evolve this system according to the Hamiltonian (3.1) with initial state  $|\psi_0\rangle$ ; if we choose the evolution time  $T$  large enough, a subsequent measurement then yields a solution to our problem with high probability. This algorithm is called the **Quantum Adiabatic Algorithm** (QAA). However, depending on the problem, it might take an unreasonably large evolution time to solve the problem at hand with high probability, making the algorithm potentially inefficient, and in some cases, the algorithm might even fail due to the minimum spectral gap not being strictly positive; see [42].

### 3.1.2 Examples for encodings of combinatorial optimization problems

Before we derive QAOA from QAA, let us first briefly discuss the formulation of the NP-complete problems MAX-CUT and MAX- $k$ -CUT (see Subsection 2.1.1) as problem Hamiltonians. A wide variety of formulations for, e.g., all of Karp's 21 NP-complete problems [21], can be found in [44]. Let  $\mathcal{P} = (\mathcal{I}, \mathcal{F} = \{\mathcal{F}(I)\}_{I \in \mathcal{I}}, \mathcal{S} = \{\mathcal{S}(I)\}_{I \in \mathcal{I}}, c = \{c_I\}_{I \in \mathcal{I}})$  be a combinatorial optimization problem, where we assume that for every  $I \in \mathcal{I}$ , there are  $k, n \in \mathbb{N}$  such that  $\mathcal{F}(I) = \mathbb{Z}_k^n$ . In this thesis, we require every problem Hamiltonian  $H_I^{\mathcal{P}}$  to satisfy

$$H_I^{\mathcal{P}}|z\rangle = c_I(z)|z\rangle \quad (3.2)$$

for any  $z \in \mathbb{Z}_k^n$ . This implies that the eigenspace associated with the largest eigenvalue corresponds to the solutions of the combinatorial optimization problem rather than the groundspace; the adiabatic theorem can be formulated analogously in this setting.

#### MAX-CUT

Given an undirected unweighted graph  $G = (V, E)$ , we first associate a qubit to every vertex of the graph, i.e., the state space of the quantum system is  $(\mathbb{C}^2)^{\otimes |V|}$ . We then define the MAX-CUT Hamiltonian associated with  $G$  as

$$H_G^{\text{MC}} = \frac{1}{2} \sum_{\{u,v\} \in E} (I - Z_u Z_v) . \quad (3.3)$$

The eigenspace associated with the largest eigenvalue of this Hamiltonian is equal to

$$\text{span}\{|z\rangle \in \{|0\rangle, |1\rangle\}^{\otimes |V|} : z \in \{0, 1\}^{|V|} \text{ achieves the maximum cutsize}\} .$$



### MAX- $k$ -CUT

Let  $k \geq 2$  be an integer. Given an undirected unweighted graph  $G = (V, E)$ , we first associate a  $k$ -dimensional qudit to every vertex of the graph, i.e., the state space of the quantum system is  $(\mathbb{C}^k)^{\otimes |V|}$ . We then define the MAX- $k$ -CUT Hamiltonian associated with  $G$  as

$$H_G^{\text{MkC}} = \frac{1}{k} \sum_{\{u,v\} \in E} \sum_{b \in \mathbb{Z}_k} (1 - \delta_{b,0}) \Pi_{u,v}(b) ,$$

where  $\Pi_{u,v}(b) = \sum_{a \in \mathbb{Z}_k} |a, a+b \bmod k\rangle \langle a, a+b \bmod k|_{u,v}$  is a projector onto the subspace spanned by the two qudits  $u$  and  $v$ . Similarly to the MAX-CUT case, the eigenspace associated with the largest eigenvalue of this Hamiltonian is equal to

$$\text{span}\{|z\rangle \in \mathbb{Z}_k^{\otimes |V|} : z \in \mathbb{Z}_k^{|V|} \text{ achieves the maximum cutsize}\} .$$

Note that for  $k = 2$ , we obtain the MAX-CUT Hamiltonian  $H_G^{\text{MC}}$ .

Families of MAX-CUT and MAX- $k$ -CUT Hamiltonians are examples for so-called **local** Hamiltonians. Letting  $\mathcal{N}$  denote some index set, a family of Hamiltonians  $\{H_n\}_{n \in \mathcal{N}}$ , where each  $H_n$  is acting on a tensor product of  $n$  subsystems, is called local if one can write each member of the family as  $H_n = \sum_{j \in \mathcal{M}_n} H_{n,j}$ , where  $\mathcal{M}_n$  is a finite index set for every  $n \in \mathcal{N}$ , and each **local term**  $H_{n,j}$  is acting non-trivially on at most a constant number of subsystems independent of the total number of subsystems in the state space. For example, the family  $\{H_G^{\text{MC}}\}_{G \in \mathcal{G}}$ , where  $\mathcal{G}$  denotes the set of all undirected unweighted graphs, is local since we can write

$$H_G^{\text{MC}} = \sum_{e=\{u,v\} \in E} H_{G,e}^{\text{MC}} , \text{ where} \tag{3.4}$$

$$H_{G,e}^{\text{MC}} = \frac{1}{2} (I - Z_u Z_v)$$

(see (3.3)) and each  $H_{G,e}^{\text{MC}}$  acts non-trivially on exactly two subsystems independent of the size of the graph. Note that for  $e_1, e_2 \in E$ , the local terms  $H_{G,e_1}^{\text{MC}}$  and  $H_{G,e_2}^{\text{MC}}$  commute, meaning that  $\{H_G^{\text{MC}}\}_{G \in \mathcal{G}}$  is a **family of commuting, local Hamiltonians**.

We call a family of local Hamiltonians  $\ell$ -local if each of its local terms acts non-trivially on exactly  $\ell$  subsystems, where  $\ell \in \mathbb{N}$ . Given an index set  $\mathcal{N}$  and a family of 2-local Hamiltonians  $\{H_n\}_{n \in \mathcal{N}}$ , we define the **interaction graph** of  $H_n$  by associating a vertex with each subsystem of the state space and connecting two vertices via an edge if there exists a local term acting non-trivially on precisely those two subsystems. For example, the family of MAX-CUT Hamiltonians is 2-local, and the interaction graph of  $H_G^{\text{MC}}$  is the unweighted version of  $G$ .

#### 3.1.3 From QAA to the QAOA ansatz

The ansatz for QAOA is now obtained by approximating the unitary evolution that the quantum adiabatic algorithm implements. The Hamiltonian  $H_T$  defined in (3.1) is time-dependent, and as mentioned in Subsection 2.2.4, it is in general difficult to obtain the associated unitary

evolution operator if the Hamiltonian governing the evolution is time-dependent as it involves evaluating a time-ordered integral. However, for a time-independent Hamiltonian  $H_{\text{const}}$ , the evolution operator is given by  $U(t) = \exp(-iH_{\text{const}}(t - t_0))$ . If a time-dependent Hamiltonian  $H$  is continuous in  $t$ , it is approximately constant and therefore approximately equal to  $H(s)$  on any interval  $[s, s + \delta] \subset [t_0, T]$  if  $\delta$  is small. These observations suggest the approximation

$$U(T) \approx \prod_{m=1}^p \exp(-iH(t_{m-1})(t_m - t_{m-1})) \quad , \quad (3.5)$$

where  $p \in \mathbb{N}$  and  $\{t_0, t_1, \dots, t_p\} \subset [t_0, T]$  with  $t_p = T$  is a discretization of the interval  $[t_0, T]$ , i.e.,  $t_k < t_{k+1}$  for all  $k \in \mathbb{Z}_p$ . In the case of adiabatic evolution, i.e.,  $H(t) = (1 - t/T)H_M + (t/T)H_I^P$ , we therefore obtain

$$U(T) \approx \prod_{m=1}^p \exp\left(-i\left(\left(1 - \frac{t_{m-1}}{T}\right)H_M + \frac{t_{m-1}}{T}H_I^P\right)(t_m - t_{m-1})\right) \quad .$$

The final step to obtain QAOA is **Trotterization**, a technique inspired by the **Lie-Trotter product formula** [45] which implies that

$$\lim_{\delta \rightarrow 0} \|\exp((A + B)\delta) - \exp(\delta A)\exp(\delta B)\| = 0$$

for all  $A, B \in \mathcal{B}(\mathbb{C}^n)$ , where  $\delta \in \mathbb{R}$ . Applying this observation non-asymptotically with  $A := -i(1 - \frac{t_m}{T})H_M$ ,  $B := -i(\frac{t_{m-1}}{T})H_I^P$  and  $\delta := t_m - t_{m-1}$ , we therefore obtain

$$U(T) \approx \prod_{m=1}^p \exp\left(-i(t_m - t_{m-1})\left(1 - \frac{t_{m-1}}{T}\right)H_M\right) \exp\left(-i(t_m - t_{m-1})\frac{t_{m-1}}{T}H_I^P\right) \quad .$$

To state the QAOA ansatz for the qubit setting with problem Hamiltonian  $H_I^P$  with state space  $(\mathbb{C}^2)^{\otimes n}$ , where  $n \in \mathbb{N}$ , we choose the so-called **transversal field** as the mixing Hamiltonian, i.e.,

$$H_M = \sum_{u \in [n]} X_u \quad ,$$

and we choose the so-called **level** of QAOA, i.e., a number  $p \in \mathbb{N}$  which quantifies the degree of discretization in (3.5): the larger we choose  $p$ , the finer the discretization becomes. The input state for the algorithm is  $|+^n\rangle := |+\rangle^{\otimes n}$  (see (3.1)) which is the unique eigenstate associated with the largest eigenvalue of  $H_M$ . Furthermore, we choose so-called **angles**  $\beta, \gamma \in \mathbb{R}^p$ ; these parameters represent the concrete time discretization that has been chosen. Note that in the QAOA ansatz, we have  $2p$  real parameters instead of only  $p$  such parameters as suggested by (3.5). Then the level- $p$  QAOA state  $|\psi_I^P(\beta, \gamma)\rangle$  is defined as

$$|\psi_I^P(\beta, \gamma)\rangle := U_I^P(\beta, \gamma)|+^n\rangle \quad , \quad \text{where} \quad (3.6)$$

$$U_I^P(\beta, \gamma) := \prod_{m=1}^p [\exp(-i\beta_m H_M) \exp(-i\gamma_m H_I^P)] \quad . \quad (3.7)$$

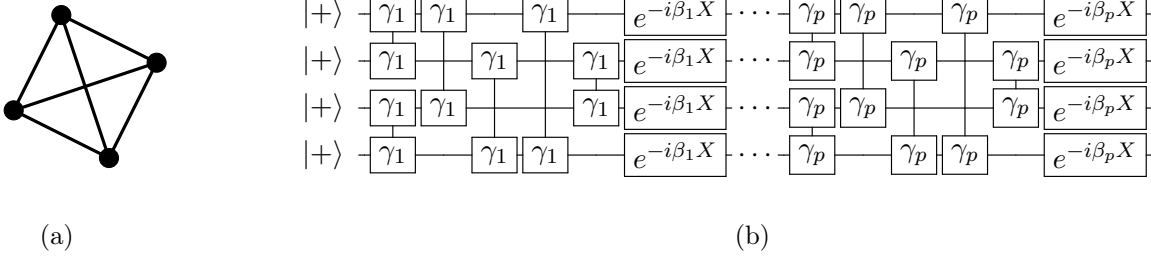


Figure 3.1: (a)  $K_4 = (V, E) := (\{1, 2, 3, 4\}, \binom{[4]}{2})$ , the unweighted complete graph on four vertices. (b) Circuit diagram associated with the QAOA unitary  $U_{K_4}^{\text{MC}}(\beta, \gamma)$ . Since  $|V| = 4$ , the circuit acts on four qubits and each QAOA layer contains four one-qubit gates (single boxes labeled with  $e^{-i\beta_k X}$ , where  $k \in \{1, \dots, p\}$ ), and since  $|E| = 6$ , each QAOA layer contains six two-qubit gates (two connected boxes, both labeled with  $\gamma_k$ , where  $k \in \{1, \dots, p\}$ ).

The unitary operator  $U_I^{\text{P}}(\beta, \gamma)$  is called the **QAOA unitary**. As an example, consider the application of this unitary operator for the MAX-CUT problem, i.e., we choose the problem Hamiltonian  $H_G^{\text{MC}} = \sum_{\{u,v\} \in E} H_{G,\{u,v\}}^{\text{MC}}$  as defined in (3.4) for an unweighted undirected graph  $G = (V, E)$ . This Hamiltonian acts on the state space  $(\mathbb{C}^2)^{\otimes |V|}$ , so the mixing Hamiltonian is given by  $H_M = \sum_{u \in V} X_u$ . Since the local terms of the problem Hamiltonian  $H_{G,e_1}^{\text{MC}}$  and  $H_{G,e_2}^{\text{MC}}$  commute for all  $e_1, e_2 \in E$  and the local terms of the mixing Hamiltonian  $X_u$  and  $X_v$  commute for all  $u, v \in V$ , we can write

$$\begin{aligned} U_G^{\text{MC}}(\beta, \gamma) &= \prod_{m=1}^p [\exp(-i\beta_m H_M) \exp(-i\gamma_m H_G^{\text{MC}})] \\ &= \prod_{m=1}^p \left[ \exp\left(-i\beta_m \sum_{u \in V} X_u\right) \exp\left(-i\gamma_m \sum_{\{u,v\} \in E} H_{G,\{u,v\}}^{\text{MC}}\right) \right] \\ &= \prod_{m=1}^p \left[ \prod_{u \in V} \exp(-i\beta_m X_u) \prod_{\{u,v\} \in E} \exp(-i\gamma_m H_{G,\{u,v\}}^{\text{MC}}) \right]. \end{aligned}$$

The operators in the product  $\prod_{u \in V} \exp(-i\beta_m X_u)$  all act on single qubits while the operators in the product  $\prod_{\{u,v\} \in E} \exp(-i\gamma_m H_{G,\{u,v\}}^{\text{MC}})$  all act on two qubits. The associated quantum circuit for the unweighted complete graph on four vertices is depicted in Figure (3.1).

The QAOA ansatz can also be generalized to the qudit setting with problem Hamiltonians with state space  $(\mathbb{C}^k)^{\otimes n}$ , where  $n, k \in \mathbb{N}$ ; see Core Article I) [1] for the explicit form of the ansatz and the application of QAOA to the MAX- $k$ -CUT problem.

### 3.1.4 Using QAOA for combinatorial optimization

Given  $p \in \mathbb{N}$  and angles  $\beta, \gamma \in \mathbb{R}^p$ , we can use the QAOA unitary  $U_I^{\text{P}}(\beta, \gamma)$  associated with a problem Hamiltonian  $H_I^{\text{P}}$  to define an approximation algorithm for the problem instance  $I$  with problem size  $n$  (e.g., the number of vertices in a graph) of the combinatorial optimization

1: **function**  $\text{QAOA}_p^{\mathcal{P}}(I \in \mathcal{I})$   
 2:     Determine  $(\beta^*, \gamma^*) \in \arg \max_{\beta, \gamma \in \mathbb{R}^p} \langle \psi_I^{\mathcal{P}}(\beta, \gamma) | H_I^{\mathcal{P}} | \psi_I^{\mathcal{P}}(\beta, \gamma) \rangle$   
 3:     Measure  $\psi_I^{\mathcal{P}}(\beta^*, \gamma^*)$  in the computational basis  $\rightarrow$  measurement outcome  $C$   
 4:     **return**  $C$

Figure 3.2: Pseudocode for  $\text{QAOA}_p^{\mathcal{P}}$ , where  $p \in \mathbb{N}$  and  $\mathcal{P} = (\mathcal{I}, \mathcal{F}, \mathcal{S}, c)$  is a combinatorial optimization problem. Note that in practice, Line 2 consists of finding  $\beta^*, \gamma^* \in \mathbb{R}^p$  satisfying (3.8) by using a quantum device.

problem  $\mathcal{P}$  by preparing the level- $p$  QAOA state  $U_I^{\mathcal{P}}(\beta, \gamma) | +^n \rangle$  and subsequently measuring this state in the computational basis to obtain a solution candidate  $z \in \{0, 1\}^n$ .

Now let  $p$  be the chosen level of QAOA. It is natural to use angles  $\beta^*, \gamma^* \in \mathbb{R}^p$  such that the expectation value of  $H_I^{\mathcal{P}}$  under the level- $p$  QAOA state with angles  $\beta^*, \gamma^*$  is made as large as possible, i.e., one wants to find  $\beta^*, \gamma^*$  such that

$$f_I^{\mathcal{P}}(\beta^*, \gamma^*) := \langle \psi_I^{\mathcal{P}}(\beta^*, \gamma^*) | H_I^{\mathcal{P}} | \psi_I^{\mathcal{P}}(\beta^*, \gamma^*) \rangle \approx \sup_{\beta, \gamma \in \mathbb{R}^p} \langle \psi_I^{\mathcal{P}}(\beta, \gamma) | H_I^{\mathcal{P}} | \psi_I^{\mathcal{P}}(\beta, \gamma) \rangle \quad . \quad (3.8)$$

This is motivated by (3.2), where we required problem Hamiltonians to be diagonal with respect to the computational basis; if  $|\psi_I^{\mathcal{P}}(\beta^*, \gamma^*)\rangle$  is a computational basis state that represents a solution of the problem instance, then the supremum on the right-hand side of (3.8) is in fact equal to the optimal value of the cost function since QAOA concludes with a measurement in the computational basis.

Note that the supremum on the right-hand side of (3.8) is always finite due to the Cauchy-Schwarz inequality and the fact that  $\| |\psi_I^{\mathcal{P}}(\beta, \gamma)\rangle \| = 1$ . The function  $f_I^{\mathcal{P}}$  is continuous, and if the associated problem Hamiltonian  $H_I^{\mathcal{P}}$  has only integer eigenvalues for all  $I \in \mathcal{I}$ , we have

$$f_I^{\mathcal{P}}(\beta, \gamma) = f_I^{\mathcal{P}}(\beta + k\pi, \gamma + 2\pi\ell)$$

for all  $\beta, \gamma \in \mathbb{R}^p$  and all binary,  $p$ -dimensional vectors  $k, \ell$ . This implies that  $f_I^{\mathcal{P}}(\mathbb{R}^p \times \mathbb{R}^p) = f_I^{\mathcal{P}}([0, 2\pi]^p \times [0, 2\pi]^p)$  and since  $[0, 2\pi]^p \times [0, 2\pi]^p$  is compact, the supremum in (3.8) is attained on this compact set. This assumption on the eigenvalues of  $H_I^{\mathcal{P}}$  is satisfied by both unweighted MAX-CUT and MAX- $k$ -CUT problems and many other combinatorial optimization problems.

Since we will only apply QAOA to such combinatorial optimization problems in this thesis, we from now on assume that the supremum on the right-hand side of (3.8) is attained on  $[0, 2\pi]^p \times [0, 2\pi]^p$ . Therefore, the goal is to find

$$(\beta^*, \gamma^*) \in \arg \max_{\beta, \gamma \in [0, 2\pi]^p} \langle \psi_I^{\mathcal{P}}(\beta, \gamma) | H_I^{\mathcal{P}} | \psi_I^{\mathcal{P}}(\beta, \gamma) \rangle \quad . \quad (3.9)$$

We now have everything to state the algorithm  $\text{QAOA}_p^{\mathcal{P}}$ , where  $p \in \mathbb{N}$  and  $\mathcal{P}$  is a combinatorial optimization problem; see Figure 3.2.

To quantify the performance of the algorithm, we extend the definition of an approximation ratio (Definition 2.9) to  $\text{QAOA}_p^{\mathcal{P}}$  in a straightforward way.

**Definition 3.1** (Approximation Ratio of  $\text{QAOA}_p^{\mathcal{P}}$ )

Let  $\mathcal{P} = (\mathcal{I}, \mathcal{F}, \mathcal{S}, c)$  be a combinatorial optimization problem and  $p \geq 1$  be an integer.

- (i) Let  $I \in \mathcal{I}$ . Then if  $\max_{z \in \mathcal{F}(I)} c_I(z) \neq \min_{z \in \mathcal{F}(I)} c_I(z)$ , the **approximation ratio** of  $\text{QAOA}_p^{\mathcal{P}}$  for  $I$  is defined as

$$\alpha_I(\text{QAOA}_p^{\mathcal{P}}) := \frac{\sup_{\beta, \gamma \in \mathbb{R}^p} \langle \psi_I^{\mathcal{P}}(\beta, \gamma) | H_I^{\mathcal{P}} | \psi_I^{\mathcal{P}}(\beta, \gamma) \rangle - \min_{z \in \mathcal{F}(I)} c_I(z)}{\max_{z \in \mathcal{F}(I)} c_I(z) - \min_{z \in \mathcal{F}(I)} c_I(z)}$$

and as  $\alpha_I(\text{QAOA}_p^{\mathcal{P}}) = 1$  if  $\max_{z \in \mathcal{F}(I)} c_I(z) = \min_{z \in \mathcal{F}(I)} c_I(z)$ .

- (ii) Let  $\mathcal{J} \subset \mathcal{I}$ . Then the **worst-case approximation ratio** of  $\text{QAOA}_p^{\mathcal{P}}$  on  $\mathcal{J}$  is defined as

$$\alpha_{\mathcal{J}}(\text{QAOA}_p^{\mathcal{P}}) = \inf_{I \in \mathcal{J}} \alpha_I(\text{QAOA}_p^{\mathcal{P}}) \quad .$$

If  $\mathcal{J} = \mathcal{I}$ , we write  $\alpha(\text{QAOA}_p^{\mathcal{P}})$  instead of  $\alpha_{\mathcal{J}}(\text{QAOA}_p^{\mathcal{P}})$  and call this quantity the **worst-case approximation ratio of  $\text{QAOA}_p^{\mathcal{P}}$** .

### Finding suitable variational parameters

In this thesis, we will be mostly concerned with the performance of QAOA as quantified by its approximation ratio defined in Definition 3.1. However, this assumes that one has found optimal angles  $\beta^*, \gamma^*$  as specified in (3.9), a problem that has been shown to be NP-hard in general [46]. Therefore, optimizing this expected value is typically done using heuristic methods. In practice, a quantum device is used to find good angles in Line 2 of Figure 3.2.

One of the most popular methods to do so is **gradient descent** (see [47, 48] for introductory treatments). Given  $n \in \mathbb{N}$  and  $f: D \rightarrow \mathbb{R}$  with  $D \subset \mathbb{R}^n$  being compact and  $f$  being smooth on the interior of  $D$ , the goal of gradient descent is to find an element of  $\arg \max_{x \in D} f(x)$ . Gradient descent is an iterative procedure, i.e., it produces a sequence  $(x_k)_{k \in \mathbb{N}}$  such that hopefully,  $f(x_k) \approx \max_{x \in D} f(x)$  for sufficiently large  $k$ . For gradient descent, this sequence is defined by

$$\begin{aligned} x_0 &\in D \quad , \\ x_{k+1} &:= x_k + \gamma_k \nabla f(x_k) \quad , \end{aligned}$$

where  $\gamma_k \in \mathbb{R}_{>0}$  is a suitably chosen stepsize. While there are sufficient criteria under which gradient descent converges to a global maximum (e.g., concavity), in general, convergence of the method is not guaranteed. In particular, if  $f$  is not concave, it may have several local, but non-global maxima, in which case the method might converge to such a local maximum instead. Since the function  $(\beta, \gamma) \mapsto f_I^{\mathcal{P}}(\beta, \gamma) = \langle \psi_I^{\mathcal{P}}(\beta, \gamma) | H_I^{\mathcal{P}} | \psi_I^{\mathcal{P}}(\beta, \gamma) \rangle$  is not concave, this may happen for the optimization within QAOA. Another obstacle is a well known phenomenon known as ‘‘barren plateau’’ [49]: the optimization may get stuck in a ‘‘flat’’ region of  $\mathbb{R}^{2p}$ , i.e., a region for which

$\nabla f_I^{\mathcal{P}} \approx 0$ . This problem has been extensively studied in several works [50–53] where techniques to avoid such convergence issues are explored. Furthermore, each step of gradient descent requires an evaluation of the cost function as well as the computation and evaluation of the gradient of the cost function, both of which are done using the quantum device in practice. This has the consequence that the larger the chosen level  $p$ , the more expensive these evaluations become and the more computationally intractable gradient descent becomes.

A more primitive heuristic to find suitable angles is **grid search**, where one chooses a finite subset  $T \subset \mathbb{R}$  and determines  $\arg \max_{(\beta, \gamma) \in T^p \times T^p} f_I^{\mathcal{P}}(\beta, \gamma)$  to approximate the maximum. Since  $|T^p \times T^p|$  grows exponentially with the level  $p$ , this method is only tractable if  $p$  is very small.

While gradient descent and grid search are two heuristic methods for the optimization within QAOA, there are in fact classes of problems for which one can find analytic expressions for the optimal angles. For example, [54] gives an explicit expression for the optimal angles for QAOA<sub>1</sub><sup>MC</sup> on triangle-free regular graphs.

Finally, it was numerically argued in [55] for MAX-CUT that if the angles  $\beta, \gamma$  are fixed and problem instances  $G \in \mathcal{D}_3$  are sampled from a sufficiently well-behaved distribution of 3-regular graphs, then the value of  $f_G^{\text{MC}}(\beta, \gamma)$  is almost constant independent of the instance  $G$ . This implies that once one has found optimal or near-optimal angles for a typical instance, they will with high probability also yield high expectation values for other, randomly drawn instances.

## 3.2 Properties of the QAOA ansatz

### 3.2.1 Locality

Let  $\mathcal{P}$  be a combinatorial optimization problem with instance set  $\mathcal{I}$  and assume that the family of problem Hamiltonians  $\{H_I^{\mathcal{P}}\}_{I \in \mathcal{I}}$  is a family of 2-local Hamiltonians, so we can write  $H_I^{\mathcal{P}} = \sum_{e \in E} H_e^{\mathcal{P}}$  with 2-local terms  $H_e$ , where  $G = (V, E)$  is the interaction graph of  $H_I^{\mathcal{P}}$ . The locality of  $H_I^{\mathcal{P}}$  implies that its expected value under a QAOA ansatz state  $|\psi_I^{\mathcal{P}}(\beta, \gamma)\rangle$  can be written as

$$\langle \psi_I^{\mathcal{P}}(\beta, \gamma) | H_I^{\mathcal{P}} | \psi_I^{\mathcal{P}}(\beta, \gamma) \rangle = \sum_{e \in E} \langle \psi_I^{\mathcal{P}}(\beta, \gamma) | H_e^{\mathcal{P}} | \psi_I^{\mathcal{P}}(\beta, \gamma) \rangle. \quad (3.10)$$

Focusing on a single term  $H_e^{\mathcal{P}}$ , one can see that  $\langle \psi_I^{\mathcal{P}}(\beta, \gamma) | H_e^{\mathcal{P}} | \psi_I^{\mathcal{P}}(\beta, \gamma) \rangle$  does not depend on all subsystems of the state space, but only on a certain so-called  **$p$ -neighborhood** of  $e$ ; QAOA <sub>$p$</sub>  is therefore a **local** ansatz. This property follows from the fact that terms in the QAOA <sub>$p$</sub>  unitary (3.7) which do not have support on this  $p$ -neighborhood commute through  $H_e^{\mathcal{P}}$  and cancel with their adjoints since they are unitary [12]. For the MAX-CUT Hamiltonian, such  $p$ -neighborhoods are illustrated in Figure 3.3.

While we illustrated this property in terms of the expectation value (3.10), it can be formulated more generally for other local operators and  $\ell$ -local operators where  $\ell \neq 2$ , see Section 3.2 of Core Article II) [2].

The locality of QAOA allows to numerically evaluate expectation values such as (3.10) by a classical algorithm as long as  $p$  is a small constant and the interaction graph has a small maximum degree. In that case, only a small constant number of subsystems actually contributes nontrivially

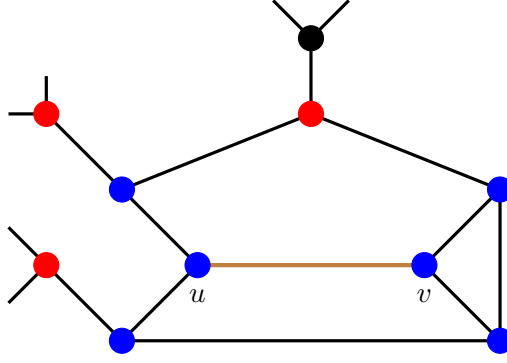


Figure 3.3: Illustration of 1- and 2-neighborhoods of an edge  $e = \{u, v\}$  in an interaction graph. The 1-neighborhood of  $e$  consists of all vertices colored blue since each of these vertices is either incident to  $u$  or  $v$ . The 2-neighborhood of  $e$  consists of all vertices colored blue or red since each of these vertices is either an element of the 1-neighborhood of  $\{u, v\}$  or is incident to an element in the 1-neighborhood. The vertex colored black is neither of those and is therefore not an element of either of the two neighborhoods.

to each summand, so the dimension of the corresponding Hilbert space is small enough such that a classical evaluation is computationally tractable. The locality of QAOA has been exploited previously in the literature [12, 56] to establish lower bounds on the approximation ratio, and we also heavily make use of this technique in Core Article II) [2]. However, we emphasize that in practice, a quantum device would be used to evaluate (3.10) in the angle optimization of QAOA.

### 3.2.2 Uniformity

Again, let  $\mathcal{P}$  be a combinatorial optimization problem with instance set  $\mathcal{I}$  and assume again that the family of problem Hamiltonians  $\{H_I^{\mathcal{P}}\}_{I \in \mathcal{I}}$  is a family of 2-local Hamiltonians, so we can write  $H_I^{\mathcal{P}} = \sum_{e \in E} H_e^{\mathcal{P}}$  with 2-local terms  $H_e$ , where  $G = (V, E)$  is the interaction graph of  $H_I^{\mathcal{P}}$ . Now let  $e_1, e_2 \in E$  be two edges such that the  $p$ -neighborhoods of  $e_1$  and  $e_2$  are isomorphic, i.e., there is an edge-preserving bijection between their vertex sets. It follows immediately from the definition of  $\text{QAOA}_p$  that then

$$\langle \psi_I^{\mathcal{P}}(\beta, \gamma) | H_{e_1}^{\mathcal{P}} | \psi_I^{\mathcal{P}}(\beta, \gamma) \rangle = \langle \psi_I^{\mathcal{P}}(\beta, \gamma) | H_{e_2}^{\mathcal{P}} | \psi_I^{\mathcal{P}}(\beta, \gamma) \rangle$$

holds, meaning that the QAOA ansatz is **uniform**: for fixed angles  $\beta, \gamma$ , the values of the summands in (3.10) only depend on the isomorphism classes of the  $p$ -neighborhoods of the edge associated with the summand.

Focusing on the unweighted MAX-CUT problem, the uniformity of QAOA means that one can, e.g., rewrite the expectation value of  $H_G^{\text{MC}}$  of a QAOA state  $|\psi_G^{\text{MC}}(\beta, \gamma)\rangle$  as

$$\langle \psi_G^{\text{MC}}(\beta, \gamma) | H_G^{\text{MC}} | \psi_G^{\text{MC}}(\beta, \gamma) \rangle = \sum_{S \in \mathcal{N}} n_S(G) \langle \psi_S^{\text{MC}}(\beta, \gamma) | H_S^{\text{MC}} | \psi_S^{\text{MC}}(\beta, \gamma) \rangle, \quad (3.11)$$

where  $\mathcal{N}$  is the set of isomorphism classes of  $p$ -neighborhoods,  $n_S(G)$  denotes the number of times that an edge  $e \in E$  has the  $p$ -neighborhood  $S$  and  $H_S^{\text{MC}}$  is the Hamiltonian acting on the subgraph  $S$ . For small values of  $p$  and bounded-degree graphs, this property allows us to simply

list all possible  $p$ -neighborhoods, classically evaluate the cost functions on these neighborhoods and then evaluate the expected value using (3.11) by counting the number of occurrences of each isomorphism class in  $G$ . For larger values of  $p$ , there may be too many  $p$ -neighborhoods which are too large, preventing a brute-force enumeration of such values. For a demonstration of this technique, see [56].

The uniformity of QAOA also motivates to study the performance of  $\text{QAOA}_p$  on graphs of high girth, i.e., graphs which do not have cycles with less than  $2p + 2$  pairwise distinct entries. For such graphs, all  $p$ -neighborhoods are isomorphic to the same symmetric tree (e.g.,  $G_1$  in Figure 3 of Core Article II) [2] for  $p = 1$ ). Exploiting the symmetry of the tree, one can use tensor network methods to evaluate the expected value of a local term of a QAOA state on such a tree which allows to compute expectation values for such graphs; we use such techniques in Core Article II) [2]. There, one can also find a more general treatment of uniformity for general local operators.

### 3.2.3 Circuit Depth for two-local Hamiltonians

For 2-local Hamiltonians such as the MAX- $k$ -CUT Hamiltonians, the circuit depth of the QAOA circuit depends linearly on the product of two quantities: the chosen level  $p$  and the maximum degree of the underlying interaction graph. This follows from the commutativity of the local terms in the Hamiltonians and the fact that we can arrange the associated gates according to edge colorings, see [3] for a more detailed argument for MAX-CUT.

This depth dependency of the circuit on the level is one of the main points that make QAOA interesting for the NISQ era: one can choose the level of QAOA depending on the capabilities of the real-world quantum device at hand.

## 3.3 Performance guarantees for QAOA

As a first demonstration of the capabilities of QAOA, it was shown in the seminal paper [12] that applied to the MAX-CUT problem restricted to 3-regular graphs,  $\text{QAOA}_1^{\text{MC}}$  achieves a worst-case approximation ratio of  $\alpha_{\mathcal{D}_3}(\text{QAOA}_1^{\text{MC}}) \geq 0.6924$  which has been confirmed to be tight in [56]. Compared to classical algorithms (see Subsection 2.1.4),  $\text{QAOA}_1^{\text{MC}}$  provides a significant improvement over the trivial algorithm with  $\alpha_{\mathcal{D}_3}(\mathcal{T}) = 0.5$ , but does not compare favorably in terms of provable lower bounds to the Goemans-Williamson algorithm with  $\alpha_{\mathcal{D}_3}(\text{GW}) \geq 0.878$  or the HLZ algorithm with  $\alpha_{\mathcal{D}_3}(\text{HLZ}) \geq 0.9326$ .

The proof of the result for  $\text{QAOA}_1^{\text{MC}}$  exploits the locality and uniformity of QAOA. A similar analysis was then used in subsequent work to establish provable lower bounds on approximation ratios for  $p > 1$  [56]; there, the authors show that  $\alpha_{\mathcal{D}_3}(\text{QAOA}_2^{\text{MC}}) \geq 0.7559$  and conjecture  $\alpha_{\mathcal{D}_3}(\text{QAOA}_3^{\text{MC}}) \geq 0.7923$ . Furthermore, this technique was also applied to other combinatorial optimization problems [57].

From the definition of QAOA, it is clear that given a problem instance  $x$  of a combinatorial optimization problem  $\mathcal{P}$ , we have  $\alpha_x(\text{QAOA}_{p+1}^{\mathcal{P}}) \geq \alpha_x(\text{QAOA}_p^{\mathcal{P}})$  for any  $p \in \mathbb{N}$ . Furthermore, for problems such as MAX-CUT, it is known that  $\lim_{p \rightarrow \infty} \alpha_x(\text{QAOA}_p^{\mathcal{P}}) = 1$  [12], i.e.,  $\text{QAOA}_p^{\text{MC}}$  achieves the actual optimum (i.e., an approximation ratio of 1) if one chooses  $p$  sufficiently large.



## Chapter 4

# Performance limitations of QAOA

For  $\text{QAOA}_p$  applied to the MAX-CUT problem, an arbitrarily high approximation ratio can be achieved by choosing the level  $p$  large enough [12]. However, it might happen that the circuit depth (which depends linearly on  $p$ ) needed to achieve a result deemed sufficiently accurate might grow with the number of vertices in the given graph, in the worst case even exponentially as is the case for the runtime of classical brute-force algorithms. Furthermore, a larger level of QAOA implies a higher dimensionality of the parameter space which makes optimization methods such as gradient descent computationally impractical.

We therefore hope for good performance of  $\text{QAOA}_p$  with low, ideally constant values of  $p$ , especially in the light of the NISQ era. In this chapter, however, we will outline two of our results [1,3] which imply that for constant  $p$ , the worst-case approximation ratios of  $\text{QAOA}_p$  for both the MAX-CUT problem as well as the MAX- $k$ -CUT problem are upper bounded by constants which are smaller than the worst-case approximation ratios of the best classical algorithms; in order to surpass this threshold,  $p$  has to grow at least logarithmically in the number of vertices in the given graph.

### 4.1 Limitations obtained by exploiting locality and symmetry

In [3], we showed the following limitations of  $\text{QAOA}_p$  applied to MAX-CUT.

**Theorem 4.1** (Limitations of QAOA for MAX-CUT [3])

Let  $D \geq 3$  be an integer. Then there exists an infinite family of bipartite  $D$ -regular graphs  $\mathcal{G}^D = \{G_n^D\}_{n \in \mathcal{N}_D}$ , where  $\mathcal{N}_D \subset \mathbb{N}$  and  $G_n^D$  is a graph on  $n$  vertices, such that if  $p < \frac{\frac{1}{3} \log_2(n) - 4}{D+1}$ , we have

$$\alpha_{G_n^D}(\text{QAOA}_p^{\text{MC}}) \leq \frac{5}{6} + \frac{\sqrt{D-1}}{3D} \quad . \quad (4.1)$$

In particular,

$$\alpha(\text{QAOA}_p^{\text{MC}}) \leq \frac{5}{6} + \varepsilon \quad (4.2)$$

for any fixed  $p \in \mathbb{N}$  and any  $\varepsilon > 0$ .

Equation (4.2) follows directly from (4.1) by taking the limit  $D \rightarrow \infty$ . Apart from special, heavily restricted cases such as the “ring of disagrees” [58], i.e., the application of QAOA $_p$  to the MAX-CUT problem on 2-regular graphs, this was – to the best of our knowledge – the first result giving a provable upper bound on the achievable approximation ratio of QAOA $_p$  for MAX-CUT for  $p > 1$ . It quantitatively shows that for a constant level  $p$ , one always has MAX-CUT problem instances in which QAOA performs worse than the best classical algorithm by Goemans and Williamson [17], since  $\frac{5}{6} \approx 0.8333 < \alpha(\text{GW}) \approx 0.8785$  (see Subsection 2.1.4).

The key properties of QAOA that lead to this result are the symmetry and locality of both the input state  $|+^n\rangle$  (which is an eigenvector of  $X^{\otimes n}$  associated with eigenvalue  $+1$ ) as well as the QAOA unitary  $U(\beta, \gamma)$  for MAX-CUT which commutes with  $X^{\otimes n}$ , where  $n$  is the number of vertices of the graph. In both cases, we say that  $\mathbb{Z}_2$ -symmetry is satisfied. In particular, these relations are true for any choice of angles  $\beta, \gamma$ , meaning that we do not have to deal with finding optimal angles or establishing that particular angles are indeed optimal.

The most important step in the proof is to establish the fact that there exists a family of bipartite,  $D$ -regular graphs  $\mathcal{G}^D = \{G_n^D\}_{n \in \mathcal{N}_D}$  on  $n$  vertices such that the MAX-CUT Hamiltonian associated with those graphs has ground state energy 0, but their energy density  $\langle \psi | H_{G_n^D}^{\text{MC}} | \psi \rangle / n$  is lower bounded by a constant for sufficiently large  $n$  if the state  $|\psi\rangle$  was obtained by applying a low-depth,  $\mathbb{Z}_2$ -symmetric circuit to a  $\mathbb{Z}_2$ -symmetric product state.

This result is reminiscent of and motivated by the so-called **NLTS conjecture** [59], where NLTS is an abbreviation for “No Low-Energy Trivial States”. There, the goal was to show the existence of a constant  $\varepsilon > 0$  and a family of local Hamiltonians  $\{H_n\}_n$  on  $n$  qubits with ground state energy 0 such that for sufficiently large  $n$ , any  $n$ -qubit state  $|\psi_n\rangle$  that has an energy density lower than  $\varepsilon$  cannot be obtained by applying a constant-depth circuit consisting of one- and two-qubit gates to a product state. Note that the result of [60], where the NLTS conjecture was finally positively resolved, does not imply our result or vice-versa. The positive resolution of the NLTS conjecture was a significant result in complexity theory as its validity was known to be necessary for a potential quantum analogue of the classical PCP theorem [31, 32] to hold [61].

A key ingredient in establishing our version of the NLTS conjecture restricted to  $\mathbb{Z}_2$ -symmetric circuits is the following upper bound on the separation of output sets obtained from shallow quantum circuits by Eldar and Harrow:

**Lemma 4.2** (Separation of Output Sets from Shallow Quantum Circuits [62])

Let  $d \in \mathbb{N}$ ,  $U$  be an  $n$ -qubit unitary implemented by a depth- $d$  quantum circuit consisting of one- and two-qubit gates,  $|\varphi\rangle$  be an  $n$ -qubit product state and define  $p : \{0, 1\}^n \rightarrow [0, 1]$ ,  $x \mapsto p(x) = |\langle x | U | \varphi \rangle|^2$  and  $p(T) = \sum_{x \in T} p(x)$  for  $T \subseteq \{0, 1\}^n$ . Then

$$\text{dist}(S, S') \leq \frac{4\sqrt{n \cdot 2^{3d}}}{\min\{p(S), p(S')\}}$$

for any  $S, S' \subseteq \{0, 1\}^n$ , where  $\text{dist}(T_1, T_2) := \min_{x \in T_1, y \in T_2} |x - y|_H$ , and  $|z|_H$  is the Hamming weight of  $z \in \{0, 1\}^n$ .

This result implies that output distributions of low-depth quantum circuits cannot assign a non-negligible probability weight to subsets of bit strings which are highly separated with respect to Hamming weight. Since the MAX-CUT Hamiltonians are  $\mathbb{Z}_2$ -symmetric (which can be seen from the fact that flipping the bits in a given cut does not change the cutsize), we can choose two bitflip-symmetric sets which are far apart to give a logarithmic lower bound on the level of QAOA needed to achieve a certain approximation ratio; for further details, see Article III) [3]. We note that the bound in Lemma 4.2 has been further investigated and strengthened in [63] and was also used in the proof of the NLTS conjecture [60]. Furthermore, our proof technique to establish Theorem 4.1 has been further applied in the literature [64].

## 4.2 Limitations obtained by exploiting locality and uniformity

Shortly after the publication of [3], the authors of [13] showed that the worst-case performance of QAOA is even worse than shown in our work: choosing  $D \in \mathbb{N}$  large enough,  $\text{QAOA}_p$  with  $p = O(\log(n))$  can only achieve an expected approximation ratio of  $\frac{1}{2}$  for the MAX-CUT problem on random bipartite  $D$ -regular graphs with  $n$  vertices, i.e., the algorithm performs no better than the trivial algorithm for MAX-CUT described in Section 2.1.4. We note, however, that our result in [3] not only holds for QAOA, but can be generalized for any variational quantum circuits exhibiting  $\mathbb{Z}_2$ -symmetry; furthermore, we state an explicit family of graphs for which our upper bounds hold.

In [1], we use the proof technique of [13] to show the following analogous no-go results for the MAX- $k$ -CUT problem introduced in Section 2.1.4.

**Theorem 4.3** (Limitations of QAOA for MAX- $k$ -CUT [1])

Let  $k \geq 2$  be an integer. For  $n, d \in \mathbb{N}$ , denote the set of bipartite  $d$ -regular graphs on  $n$  vertices by  $\mathcal{G}_{n,d}^{\text{bi}}$ . Then there exist a constant  $\zeta > 0$  and two functions  $f, g: \mathbb{N} \rightarrow \mathbb{R}_{>0}$  with  $f(d), g(n) \in o(1)$  such that

$$\Pr_G \left[ \alpha_G \left( \text{QAOA}_p^{\text{MkC}} \right) \geq \left( 1 - \frac{1}{k} \right) + f(d) + g(n) \right] = o(1)$$

for all  $d \geq \zeta$  and  $d = o(\sqrt{n})$  and all levels  $p < \frac{1}{2} \log_d(n)$ , where the probability is taken with respect to the uniform distribution on  $\mathcal{G}_{n,d}^{\text{bi}}$ .

This result implies in particular that for suitably chosen  $(n, d, p)$ , there are graphs in  $\mathcal{G}_{n,d}^{\text{bi}}$  for which  $\text{QAOA}_p$  is incapable of achieving an approximation ratio that significantly outperforms the trivial algorithm of random guessing.

The proof of this result exploits two key characteristics of QAOA, namely its locality and uniformity (see Subsections 3.2.1 and 3.2.2). The central observation of [13] is that for uniformly random  $d$ -regular graphs on  $n$  vertices, where  $d$  and  $n$  are suitably chosen, most  $p$ -neighborhoods (see Subsection 3.2.1) in those graphs are isomorphic to the same tree with high probability, and moreover, this observation also holds for the corresponding ensemble of bipartite,  $d$ -regular graphs. By uniformity, this allows us to focus on trees as subgraphs. We then observe that for

graphs whose  $p$ -neighborhoods almost all look like a tree, the choice of optimal angles for one such graph will be nearly optimal for any other such graph. Finally, we obtain the result by exploiting a known upper bound on the typical maximum  $k$ -cutsizes of graphs in such ensembles and by using that for bipartite,  $d$ -regular graphs on  $n$  vertices, the optimal  $k$ -cutsizes is equal to the number of edges, i.e., equal to  $nd/2$ . For the precise argument, we refer to Core Article I) [1].

### 4.3 Further known limitations of QAOA in the literature

As already mentioned, the very first limitations of QAOA were shown in [58], where the authors looked at the performance of  $\text{QAOA}_p$  applied to the MAX-CUT problem on the family of cycle graphs  $\mathcal{G} = \{G_n\}_{n \in \mathcal{N}}$ , where  $\mathcal{N}$  is the set of even natural numbers and  $G_n = (\mathbb{Z}_n, \{\{k, k+1\} : k \in \mathbb{Z}_n\})$ . In this case, a simple classical efficient greedy procedure achieves optimality while  $\alpha_{G_n}(\text{QAOA}_p^{\text{MC}}) \leq \frac{2p+1}{2p+2}$  if  $p \leq \frac{n}{2}$ .

Furthermore, the authors of [56] use a simple argument exploiting the locality of QAOA to show that  $\alpha(\text{QAOA}_p^{\text{MC}}) \leq \frac{2p+2}{2p+3}$  by observing its behaviour on graphs without small cycles. In particular, it follows that  $\alpha(\text{QAOA}_p^{\text{MC}}) \leq \alpha(\text{GW})$  for  $p < 6$ .

On the more heuristic side, Hastings [65] studied the performance of local classical algorithms and found evidence that such algorithms outperform  $\text{QAOA}_1$  for virtually every regularity of graphs. This analysis was then extended in [66] to  $\text{QAOA}_2$  and has since then spawned more work in that direction, resulting in both negative and positive evidence for the performance of QAOA [67].

All of the results established in this chapter assume a perfect, noiseless execution of the QAOA circuits. This is motivated by the fact that by assuming a constant level of QAOA and bounded degree of the underlying graphs, the depth of the circuit is constant as well. However, it was found in [68] that even for low levels of QAOA, decoherence and noise will drive the output state of QAOA to a Gibbs state that is efficiently simulable on a classical computer.

## Chapter 5

# Enhancements of QAOA

In the previous chapter, we outlined several results showing that QAOA has severe shortcomings in terms of worst-case approximation ratios for both MAX-CUT and MAX- $k$ -CUT compared to known classical approximation algorithms. Naturally, one may ask how to circumvent these limitations. In this chapter, we propose two approaches towards enhancing QAOA: a recursive version of QAOA called “recursive QAOA” which was introduced in [3] and further investigated in Core Article I) [1], and a post-processed version of QAOA called “twisted QAOA” which was proposed in Core Article II) [2].

### 5.1 Recursive QAOA

In [3] and [13], the locality of QAOA was exploited to demonstrate its shortcomings in terms of approximation ratios. A natural approach to sidestep these limitations is to make the algorithm more non-local. For this purpose, we introduced a recursive version of QAOA called “RQAOA” in [3] which uses QAOA as a subroutine. This algorithm was then further investigated in Core Article I) [1] where it is also stated in its most general form. We explain the procedure applied to the so-called **Ising model** [69] which we will abbreviate by Is. This is a combinatorial optimization problem specified via the 4-tuple  $\text{Is} = (\mathcal{I}, \mathcal{F} = \{\mathcal{F}(I)\}_{I \in \mathcal{I}}, \mathcal{S} = \{\mathcal{S}(I)\}_{I \in \mathcal{I}}, c = \{c_I\}_{I \in \mathcal{I}})$ , where

$$\begin{aligned} \mathcal{I} &:= \{G: G = (V, E, w) \text{ undirected, weighted graph with } w: E \rightarrow \mathbb{Z}\} \quad , \\ \text{for } G \in \mathcal{I}, \quad \mathcal{F}(G) &:= \{0, 1\}^{|V|} \quad , \\ c_G &: \mathcal{F}(G) \rightarrow \mathbb{R} \quad , \\ x &\mapsto \sum_{e=\{u,v\} \in E} w(e) \delta_{x_u, x_v} \quad . \end{aligned}$$

Given a graph  $G = (V, E, w)$ , the associated problem Hamiltonian is given by

$$H_G^{\text{Is}} = \sum_{e=\{u,v\} \in E} w(e) Z_u Z_v \quad .$$

It is natural to analyze the performance of a quantum approximation algorithm applied to the Ising model since many important problem Hamiltonians can be expressed in terms of  $H_G^{\text{Is}}$ ; for example, we have for MAX-CUT

$$H_G^{\text{MC}} = \frac{1}{2} (|E| I - H_G^{\text{Is}})$$

for any unweighted graph  $G = (V, E)$ .

Let  $p \geq 1$  be the level of QAOA used in the procedure. On a high level,  $\text{RQAOA}_p^{\text{Is}}$  works as follows: we use the quantum device and apply  $\text{QAOA}_p^{\text{Is}}$  to determine pairs of vertices which are highly correlated or anti-correlated by computing

$$M_{u,v} := \langle \psi_G^{\text{Is}}(\beta^*, \gamma^*) | Z_u Z_v | \psi_G^{\text{Is}}(\beta^*, \gamma^*) \rangle$$

for every edge  $\{u, v\} \in E$ , where  $|\psi_G^{\text{Is}}(\beta^*, \gamma^*)\rangle$  is the level- $p$  QAOA state as defined in (3.6) with optimal angles  $\beta^*, \gamma^* \in \mathbb{R}^p$  as defined in (3.9). Next, one determines an edge

$$\{u^*, v^*\} \in \arg \max_{\{u,v\} \in E} |M_{u,v}|$$

and then substitutes

$$Z_{v^*} = \text{sgn}(M_{u^*, v^*}) Z_{u^*} \tag{5.1}$$

into  $H_G^{\text{Is}}$ . This eliminates the variable  $v^*$  from the Hamiltonian at the cost of an additional, additive identity term in the resulting Hamiltonian which only contributes an additive constant to the cost function. The resulting interaction graph then contains one fewer vertex.

This procedure (preparing the QAOA state on the current interaction graph, determining the strongest correlation/anticorrelation, eliminating a variable to obtain a new Hamiltonian with smaller associated interaction graph) is then repeated, see Figure 5.1 for an illustration of two successive variable elimination steps. We stop when the resulting interaction graph is sufficiently small such that the problem can be solved directly via, e.g., brute-force. After that, one obtains a candidate solution to the problem on the original graph by iteratively labeling the vertices according to (5.1), see Figure 5.2.

For the set of cycle graphs, we obtain the following result in [3]:

**Theorem 5.1** ( $\text{RQAOA}_1^{\text{Is}}$  on Cycle Graphs [3])

Let  $\mathcal{G} := \{G_n = (V_n, E_n, w_n)\}_{n \in \mathbb{N}}$  be an infinite family of undirected, weighted graphs with  $V_n = \mathbb{Z}_n$ ,  $E_n = \{\{k, k+1 \pmod n\} : k \in \mathbb{Z}_n\}$  and  $w_n : E_n \rightarrow \{-1, 1\}$ . Then

$$\alpha_{\mathcal{G}}(\text{RQAOA}_1^{\text{Is}}) = 1 \quad .$$

Note that it was shown in [58] that for even  $n$ , we have  $\alpha_{G_n}(\text{QAOA}_p^{\text{Is}}) \leq \frac{2p+1}{2p+2}$  if  $p \leq \frac{n}{2}$ , meaning that  $\text{RQAOA}_1^{\text{Is}}$  strictly outperforms  $\text{QAOA}_p^{\text{Is}}$  in this setting. Furthermore, we note that there is a simple, classical procedure starting at one vertex which then greedily chooses the assignment of one of the two neighboring vertices and proceeds iteratively until no vertices are left; it is

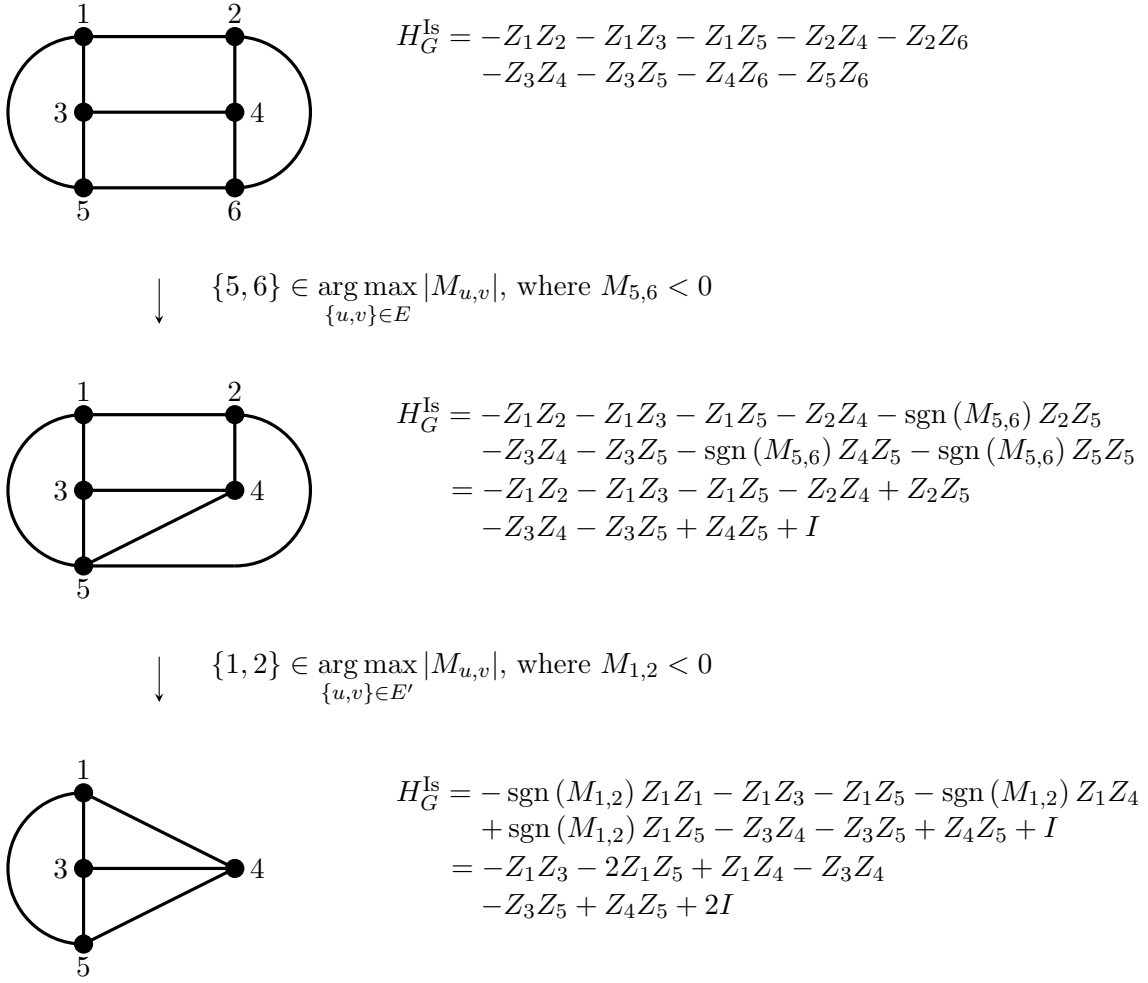


Figure 5.1: Variable elimination in RQAOA. The original graph and the associated problem Hamiltonian with  $w(e) = -1$  for each  $e \in E$  are given at the top left and right, respectively. In the first step of  $\text{RQAOA}_1^{\text{Is}}$ , the highest correlations/anticorrelations are measured on the edges  $\{1, 2\}$ ,  $\{3, 4\}$ , and  $\{5, 6\}$ , with all the pairs being anticorrelated; we choose the edge  $\{5, 6\}$  which is then contracted and leads to the vertex 6 being eliminated by an appropriate substitution into the problem Hamiltonian (middle); note the identity appearing in the new problem Hamiltonian. This procedure is repeated with the new Hamiltonian in the second step; in this case, the highest correlations/anticorrelations are measured on the edges  $\{1, 2\}$  and  $\{3, 4\}$ , with all the pairs being anticorrelated; we choose the edge  $\{1, 2\}$  which is then contracted and leads to the vertex 2 being eliminated (bottom).

easy to see that this local and efficient procedure outputs an optimal solution. To show that  $\text{RQAOA}_1^{\text{Is}}$  solves the problem perfectly, we exploit that in the variable elimination step for cycle graphs, we have only two options which permits a case distinction to determine the vertex which is eliminated next.

Unfortunately, the variable elimination step cannot be as easily analyzed when looking at general graphs. One reason for this is that during an execution of the algorithm, the maximum degree in the resulting interaction graph can grow; see e.g. Figure 5.1 where the maximum degree of the interaction graph grows by one when the first variable elimination is performed. This observation



Figure 5.2: Obtaining the candidate solution for the original graph in Figure 5.1 after the two variable eliminations specified there were performed. (a) An optimal solution for the classical problem instance associated with  $H_G^{\text{Is}} = -Z_1Z_3 - 2Z_1Z_5 + Z_1Z_4 - Z_3Z_4 - Z_3Z_5 + Z_4Z_5 + 2I$ , i.e., the final Hamiltonian in Figure 5.1. This solution was obtained via brute-force calculation. (b) The final output of  $\text{RQAOA}_1^{\text{Is}}$  applied to the original graph in Figure 5.1. We assign 0 to the vertex 2 since it was negatively correlated with vertex 1 which had been assigned the value 1 in the solution in (a). After that, we assign 1 to the vertex 6 since it was negatively correlated with vertex 5 which had been assigned the value 0 in the solution in (a).

has the negative consequence that the circuit depth of QAOA used in each recursive step may grow as well.

Because of the difficulty of a rigorous analysis, we study the performance of  $\text{RQAOA}$  applied to the MAX-3-CUT problem in Core Article I) [1] empirically instead of proving rigorous results. We introduce a classical simulation routine that allows us to efficiently compute expectation values of Ising type terms, i.e., terms of the form  $Z_uZ_v$ , under  $\text{QAOA}_1$  states. We then use this algorithm to empirically compare the performance of  $\text{QAOA}_1$ ,  $\text{RQAOA}_1$ , and the best known classical algorithm by Newman [14] on 3-colorable,  $d$ -regular graphs on up to  $n = 300$  vertices. We find that for some combinations of  $n$  and  $d$ ,  $\text{RQAOA}_1$  is able to outperform the classical algorithm while the converse is true for other combinations. It is an open problem to characterize the regimes for which  $\text{RQAOA}_1$  outperforms Newman’s algorithm; note, however, that  $\text{RQAOA}_1$  is itself a classical algorithm due to its aforementioned simulability. We also note that  $\text{RQAOA}_1$  consistently significantly outperforms  $\text{QAOA}_1$  in our simulations. We refer to the paper [1] for figures corroborating these claims.

$\text{RQAOA}$  has been extensively applied on real-world devices and in simulations for benchmarking in the literature (see, e.g., [70, 71]).

## 5.2 Twisted QAOA

The second proposed modification of QAOA in this thesis is inspired by classical post-processing techniques and is introduced in Core Article II) [2]. For this modification, we focus on the MAX-CUT problem on unweighted 3-regular graphs; however, the techniques of the article can be extended to bounded-degree graphs and are in principle applicable to any variational quantum algorithm, not just QAOA.



Recall from Section 2.1.4 that for 3-regular graphs, it is possible to surpass the approximation ratio of the Goemans-Williamson algorithm by an efficient classical algorithm. Both results mentioned there, i.e., the method by Feige, Karpinski and Langberg (FKL) [15] as well as the one by Halperin, Livnat, and Zwick (HLZ) [16], first solve a slightly modified semidefinite program similar to the one in the Goemans-Williamson algorithm to obtain a cut by randomized rounding before using simple post-processing techniques to increase the size of the cut even further. The key to their established approximation ratios is that for both of these techniques, there are functions  $I_G: \{0, 1\}^n \rightarrow \mathbb{R}_{\geq 0}$ , where  $G$  is a 3-regular graph on  $n$  vertices, such that

$$\text{cutsize}_G(C') \geq \text{cutsize}_G(C) + I_G(C)$$

holds, where  $\text{cutsize}_G(D)$  denotes the number of satisfied edges in a cut  $D$  of  $G$ ,  $C$  is the cut of  $G$  on which we perform the post-processing and  $C'$  is the final cut of  $G$  after the post-processing. Crucially,  $I_G$  can be explicitly efficiently computed in both cases and therefore quantifies the guaranteed improvements of these procedures. For example, for FKL, we have

$$I_G(C) = \frac{1}{3} |S_G(C)| \quad , \quad (5.2)$$

where  $S_G(C)$  is the set of all three-tuples  $(c, j, k) \in V^3$  such that  $\{c, j\}, \{c, k\} \in E$  and  $C(c) = C(j) = C(k)$ . The main idea in our proposal to improve QAOA is to realize that in both cases, we can “quantize” the improvement functions  $I_G$ ; as a result, we obtain a corresponding positive semidefinite operator  $I_G^q$  for which one can then show that

$$\mathbb{E}[\text{cutsize}(C')] = \langle \psi | (H_G^{\text{MC}} + I_G^q) | \psi \rangle \quad , \quad (5.3)$$

where  $|\psi\rangle \in (\mathbb{C}^2)^{\otimes n}$  and  $C'$  is the final cut that one obtains after applying the chosen post-processing procedure to the result that one obtains after measuring  $|\psi\rangle$  in the computational basis. For example, the quantized version of  $I_G$  in (5.2) is given by

$$I_G^q = \sum_{(c,j,k) \in T_G} \Pi_{c,j,k} \quad , \quad \text{where} \quad \Pi_{c,j,k} := (|000\rangle\langle 000| + |111\rangle\langle 111|)_{c,j,k} \quad ,$$

where  $T_G$  denotes the set of all three-tuples of distinct vertices lying on a path of length 2 (so-called triplets) in  $G$ .

We note that the corresponding operators for both procedures are local, hence the observable  $H_G^{\text{MC}} + I_G^q$  is local as well. Since we would like to maximize (5.3), we propose the following modification of QAOA: we first optimize the angles of the QAOA $_p^{\text{MC}}$  ansatz with respect to the observable  $H_G^{\text{MC}} + I_G^q$  instead of  $H_G^{\text{MC}}$  to obtain angles  $\beta^*, \gamma^* \in [0, 2\pi]^p$ . Then, we prepare the QAOA $_p^{\text{MC}}$  state  $|\psi_G^{\text{MC}}(\beta^*, \gamma^*)\rangle$  which we measure with respect to the computational basis, obtaining a cut  $C$  to which we then apply the respective post-processing procedure.

We call this algorithm “twisted QAOA”. Crucially, the circuit used in the algorithm is exactly the same as in regular QAOA, but with different angles; therefore, the algorithm uses exactly the same quantum resources. Furthermore, this algorithm is not restricted to QAOA; in principle,

any variational quantum algorithm can be used. In order to obtain provable lower bounds, however, one needs to be able to establish expressions similar to Eq. (5.3).

We can use the techniques from [12, 56] which exploit the locality of QAOA to obtain the results depicted in Figure 1 in [2]. We empirically observe that for low levels  $p$  of QAOA, the twisted versions of the algorithm perform similarly to  $\text{QAOA}_{p+1}$ , therefore effectively saving one level of QAOA and reducing both the circuit depth and the number of parameters.

### 5.3 Other proposals for enhancing QAOA in the literature

In addition to the proposals introduced in the two previous sections, there have been other suggestions for improving QAOA. Most prominently, the usage of different mixing Hamiltonians other than the transversal field Hamiltonian has been proposed in [72], an ansatz which the authors call “Quantum Alternating Operator Ansatz” (also abbreviated as QAOA since it includes the “standard QAOA” as a special case). One motivation for this proposal is the fact that for some combinatorial optimization problems, the standard QAOA is not capable of producing certain states. This may lead to inaccurate solutions of the combinatorial optimization problem. On the other hand, a different mixing Hamiltonian may be useful for combinatorial optimization problems with hard constraints such as the Maximum Independent Set problem [21], where the goal is to find a set of vertices of maximum cardinality in a given graph such that no vertices in the set share an edge. Using the standard binary encoding where membership of a vertex in a set is indicated by 1 while 0 denotes that the vertex is not an element of the set, not every computational basis state corresponds to a feasible solution of the problem. Here, a different mixing Hamiltonian could ensure that the output of the algorithm always satisfies such constraints.

Another, potentially complementary approach is to use a different initial state for QAOA instead of the uniform superposition; this approach is called “warm-starting QAOA”. In [71], the authors use a different mixing Hamiltonian and apply this version of QAOA to a product state obtained by encoding the output of a classical approximation algorithm such as the Goemans-Williamson algorithm (GW). This ansatz has the attractive feature that one is guaranteed to achieve  $\alpha$  (GW) since one can simply discard the output of the algorithm if the cut obtained by the quantum algorithm is not good enough. In another approach, the authors of [73] do not change the QAOA unitary, but apply the algorithm to a biased superposition of computational basis states corresponding to possible cuts in a given graph which have also been obtained by classical means. However, the most naive warm-start, i.e., simply using a computational basis state as input state while not changing the QAOA unitary at all, very likely does not lead to improvements of the performance of the algorithm as was argued in [74]. Furthermore, it is not expected that these proposals are capable of strictly outperforming the Goemans-Williamson algorithm in terms of worst-case approximation ratio as this would have far-reaching complexity theoretic consequences: in that case, the Unique Games Conjecture [28] would be false or  $\text{NP} \subset \text{BQP}$  which is not expected to be true.

# Bibliography

- [1] Sergey Bravyi, Alexander Kliesch, Robert Koenig, and Eugene Tang. Hybrid quantum-classical algorithms for approximate graph coloring. *Quantum*, 6:678, 2022.
- [2] Libor Caha, Alexander Kliesch, and Robert Koenig. Twisted hybrid algorithms for combinatorial optimization. *Quantum Science and Technology*, 7(045013), 2022.
- [3] Sergey Bravyi, Alexander Kliesch, Robert Koenig, and Eugene Tang. Obstacles to variational quantum optimization from symmetry protection. *Phys. Rev. Lett.*, 125(260505), 2020.
- [4] Richard P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21:467–488, 1982.
- [5] Paul Benioff. The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines. *Journal of Statistical Physics*, 22:563–591, 1980.
- [6] Frank Arute, Kunal Arya, Ryan Babbush, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574:505–510, 2019.
- [7] Philip Ball. First quantum computer to pack 100 qubits enters crowded race. *Nature*, 599(542), 2021.
- [8] Peter W. Shor. Algorithms for Quantum Computation: Discrete Logarithms and Factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994.
- [9] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, page 212–219, 1996.
- [10] Daniel Gottesman. Stabilizer Codes and Quantum Error Correction, 1997. arXiv:quant-ph/9705052.
- [11] John Preskill. Quantum Computing in the NISQ era and beyond. *Quantum*, 2:79, 2018.
- [12] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A Quantum Approximate Optimization Algorithm, 2014. arXiv:1411.4028.

## BIBLIOGRAPHY

- [13] Edward Farhi, David Gamarnik, and Sam Gutmann. The Quantum Approximate Optimization Algorithm Needs to See the Whole Graph: Worst Case Examples, 2020. arXiv:2005.08747.
- [14] Alantha Newman. Complex Semidefinite Programming and Max-k-Cut. In *1st Symposium on Simplicity in Algorithms*, volume 61, pages 13:1–13:11, 2018.
- [15] Uriel Feige, Marek Karpinski, and Michael Langberg. Improved Approximation of Max-Cut on Graphs of Bounded Degree. *J. Algorithms*, 43(2), 2002.
- [16] Eran Halperin, Dror Livnat, and Uri Zwick. MAX CUT in cubic graphs. *Journal of Algorithms*, 53(2), 2004.
- [17] Michel X. Goemans and David P. Williamson. Improved Approximation Algorithms for Maximum Cut and Satisfiability Problems Using Semidefinite Programming. *J. ACM*, 42(6), 1995.
- [18] Sanjeev Arora and Boaz Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, 2009.
- [19] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. MIT Press, 3rd edition, 2009.
- [20] Alan M. Turing. On Computable Numbers, with an Application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, s2-42(1):230–265, 1937.
- [21] Richard M. Karp. Reducibility among Combinatorial Problems. In *Complexity of Computer Computations*, pages 85 – 103, 1972.
- [22] David P. Williamson and David B. Shmoys. *The Design of Approximation Algorithms*. Cambridge University Press, 2010.
- [23] Narendra Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4), 1984.
- [24] Robert G. Bland, Donald Goldfarb, and Michael J. Todd. The Ellipsoid Method: A Survey. *Operations Research*, 29(6), 1981.
- [25] Gerald B. Folland. *A Course in Abstract Harmonic Analysis*. CRC Press, 1994.
- [26] Sanjeev Mahajan and Hariharan Ramesh. Derandomizing Semidefinite Programming Based Approximation Algorithms. In *Proceedings of IEEE 36th Annual Foundations of Computer Science*, pages 162 – 169, 1995.
- [27] Alan Frieze and Mark Jerrum. Improved Approximation Algorithms for MAX k-CUT and MAX BISECTION. *Algorithmica*, 18(1), 1997.
- [28] Subhash Khot. On the Power of Unique 2-Prover 1-Round Games. In *Proceedings of the Thiry-Fourth Annual ACM Symposium on Theory of Computing*, page 767–775, 2002.

- [29] Etienne Klerk, Dmitrii Pasechnik, and Joost P. Warners. On Approximate Graph Colouring and MAX- $k$ -CUT algorithms based on the  $\vartheta$ -function. *Journal of Combinatorial Optimization*, 8:267–294, 2004.
- [30] Michel X. Goemans and David P. Williamson. Approximation Algorithms for MAX-3-CUT and Other Problems via Complex Semidefinite Programming. *Journal of Computer and System Sciences*, 68(2), 2004.
- [31] Sanjeev Arora, Carsten Lund, Rajeev Motwani, et al. Proof Verification and the Hardness of Approximation Problems. *J. ACM*, 45(3), 1998.
- [32] Irit Dinur. The PCP Theorem by Gap Amplification. *J. ACM*, 54(3), 2007.
- [33] Johan Håstad. Some Optimal Inapproximability Results. *J. ACM*, 48(4), 2001.
- [34] Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan O’Donnell. Optimal Inapproximability Results for MAX-CUT and Other 2-Variable CSPs? *SIAM J. Comput.*, 37(1), 2007.
- [35] Elchanan Mossel, Ryan O’Donnell, and Krzysztof Oleszkiewicz. Noise stability of functions with low influences: Invariance and optimality. *Annals of Mathematics, second Series*, 171(1), 2010.
- [36] Sartaj Sahni and Teofilo Gonzalez. P-Complete Approximation Problems. *J. ACM*, 23(3), 1976.
- [37] David J. Griffiths and Darrell F. Schroeter. *Introduction to Quantum Mechanics*. Cambridge University Press, 3rd edition, 2018.
- [38] Michael Reed and Barry Simon. *Methods of Modern Mathematical Physics I: Functional Analysis*. Elsevier Science, 1981.
- [39] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.
- [40] Michael Grossman and Robert Katz. *Non-Newtonian Calculus: A self-contained, elementary exposition of the authors’ investigations*. Lee Press, 1972.
- [41] Eleanor Rieffel and Wolfgang Polak. *Quantum Computing: A Gentle Introduction*. The MIT Press, 2011.
- [42] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum Computation by Adiabatic Evolution, 2000. arXiv:quant-ph/0001106.
- [43] Albert Messiah. *Quantum Mechanics, Vol. II*. Dover Publications, 1999.
- [44] Andrew Lucas. Ising formulations of many NP problems. *Frontiers in Physics*, 2, 2014.
- [45] Hale F. Trotter. On the product of semi-groups of operators. In *Proceedings of the American Mathematical Society*, volume 10(4), pages 545 – 551, 1959.

## BIBLIOGRAPHY

- [46] Lennart Bittel and Martin Kliesch. Training Variational Quantum Algorithms is NP-hard. *Physical Review Letters*, 127(12), 2021.
- [47] Edwin K.P. Chong and Stanislaw H. Żak. *An Introduction to Optimization*. Wiley, 2013.
- [48] Stephen Boyd and Lieven Vandenbergh. *Convex Optimization*. Cambridge University Press, 2004.
- [49] Jarrod R. McClean, Matthew P. Harrigan, Masoud Mohseni, et al. Low depth mechanisms for quantum optimization, 2020. arXiv:2008.08615.
- [50] Stefan H. Sack, Raimel A. Medina, Alexios A. Michailidis, et al. Avoiding barren plateaus using classical shadows, 2022. arXiv:2201.08194.
- [51] Johannes S. Otterbach, Ricardo Manenti, Nasser Alidoust, et al. Unsupervised machine learning on a hybrid quantum computer, 2017. arXiv:1712.05771.
- [52] Jiahao Yao, Marin Bukov, and Lin Lin. Policy Gradient based Quantum Approximate Optimization Algorithm. In *Proceedings of The First Mathematical and Scientific Machine Learning Conference*, volume 107, pages 605 – 634, 2020.
- [53] Ruslan Shaydulin, Ilya Safro, and Jeffrey Larson. Multistart Methods for Quantum Approximate Optimization. In *2019 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1 – 8, 2019.
- [54] Zhihui Wang, Stuart Hadfield, Zhang Jiang, and Eleanor G. Rieffel. Quantum approximate optimization algorithm for MaxCut: A fermionic view. *Phys. Rev. A*, 97:022304, 2018.
- [55] Fernando G. S. L. Brandao, Michael Broughton, Edward Farhi, et al. For Fixed Control Parameters the Quantum Approximate Optimization Algorithm’s Objective Function Value Concentrates for Typical Instances, 2018. arXiv:1812.04170.
- [56] Jonathan Wurtz and Peter Love. MaxCut quantum approximate optimization algorithm performance guarantees for  $p > 1$ . *Phys. Rev. A*, 103:042612, 2021.
- [57] Jordi R. Weggemans, Alexander Urech, Alexander Rausch, et al. Solving correlation clustering with QAOA and a Rydberg qudit system: a full-stack approach, 2021. arXiv:2106.11672.
- [58] Glen B. Mbeng, Rosario Fazio, and Giuseppe Santoro. Quantum Annealing: a journey through Digitalization, Control, and hybrid Quantum Variational schemes, 2019. arXiv:1906.08948.
- [59] Michael H. Freedman and Matthew B. Hastings. Quantum systems on non-k-hyperfinite complexes: a generalization of classical statistical mechanics on expander graphs. *Quantum Information and Computation*, 14(1-2), 2014.
- [60] Anurag Anshu, Nikolas P. Breuckmann, and Chinmay Nirkhe. NLTS Hamiltonians from good quantum codes, 2022. arXiv:2206.13228.

- [61] Dorit Aharonov, Itai Arad, and Thomas Vidick. Guest Column: The Quantum PCP Conjecture. *SIGACT News*, 44(2), 2013.
- [62] Lior Eldar and Aram W. Harrow. Local Hamiltonians Whose Ground States are Hard to Approximate. In *58th Annual IEEE Symposium on Foundations of Computer Science*, pages 427–438, 2017.
- [63] Giacomo De Palma, Milad Marvian, Cambyse Rouzé, and Daniel Stilck França. Limitations of variational quantum algorithms: a quantum optimal transport approach, 2022. arXiv:2204.03455.
- [64] Sami Boulebnane. Improving the Quantum Approximate Optimization Algorithm with postselection, 2020. arXiv:2011.05425.
- [65] Matthew B. Hastings. Classical and Quantum Bounded Depth Approximation Algorithms, 2019. arXiv:1905.07047.
- [66] Kunal Marwaha. Local classical MAX-CUT algorithm outperforms  $p = 2$  QAOA on high-girth regular graphs. *Quantum*, 5:437, 2021.
- [67] Kunal Marwaha and Stuart Hadfield. Bounds on approximating Max  $k$ XOR with quantum and classical local algorithms. *Quantum*, 6:757, 2022.
- [68] Daniel Stilck França and Raul García-Patrón. Limitations of optimization algorithms on noisy quantum devices. *Nature Physics*, 17(11), 2021.
- [69] Hal Tasaki. *Physics and mathematics of quantum many-body systems*. Springer, 2020.
- [70] Ruslan Shaydulin and Stefan M. Wild. Exploiting symmetry reduces the cost of training qaoa. *IEEE Transactions on Quantum Engineering*, 2:1 – 9, 2021.
- [71] Daniel J. Egger, Jakub Mareček, and Stefan Woerner. Warm-starting quantum optimization. *Quantum*, 5:479, 2021.
- [72] Stuart Hadfield, Zhihui Wang, Bryan O’Gorman, et al. From the Quantum Approximate Optimization Algorithm to a Quantum Alternating Operator Ansatz. *Algorithms*, 12(2), 2019.
- [73] Reuben Tate, Majid Farhadi, Creston Herold, et al. Bridging Classical and Quantum with SDP initialized warm-starts for QAOA, 2020. arXiv:2010.14021.
- [74] Madelyn Cain, Edward Farhi, Sam Gutmann, et al. The QAOA gets stuck starting from a good classical string, 2022. arXiv:2207.05089.

BIBLIOGRAPHY



# Appendix A

## Core Articles

- A.1 Hybrid quantum-classical algorithms for approximate graph coloring

# Hybrid quantum-classical algorithms for approximate graph coloring

Sergey Bravyi, Alexander Kliesch, Robert Koenig, Eugene Tang

---

In the unweighted MAX- $k$ -CUT problem, where  $k \geq 2$  is an integer, the goal is to find a maximal  $k$ -coloring of a given graph  $G = (V, E)$ , i.e., a function  $C: V \rightarrow \mathbb{Z}_k$  that maximizes the number of edges whose endpoints have been assigned different values. In Core Article I), we study the performance of two variational quantum approximation algorithms – QAOA [12] and RQAOA [3] – applied to this NP-complete combinatorial optimization problem and compare them with classical algorithms, both heuristically and mathematically rigorously, in terms of their approximation ratios.

## A.1.1 Main Results

We first show that on the families of bipartite, regular graphs on  $n$  vertices, the probability of level- $p$  QAOA (from now on denoted by QAOA<sub>1</sub>) outperforming the trivial algorithm (where we randomly assign values to each vertex) vanishes for  $n$  going to infinity and a suitable choice of the degree of the graphs if  $p$  does not grow at least logarithmically with  $n$ .

Faced with such limitations, we then compare the performance of RQAOA with QAOA and investigate whether it performs better. To do so, we introduce a classical, time-efficient algorithm which computes the expectation value of Ising-type terms – which typically appear in the problem Hamiltonians of combinatorial optimization problems – under QAOA<sub>1</sub> output states. This allows us to numerically evaluate the performance of both QAOA<sub>1</sub> and RQAOA<sub>1</sub>; in particular, we are able to determine high-quality variational parameter in practice, making a case study possible.

We generate random,  $d$ -regular, 3-colorable connected graphs on  $n$  vertices, where  $(n, d) \in \{30, 60, 150, 300\} \times \{4, 6, 8, 10\}$ , and determine the achieved approximation algorithms of QAOA<sub>1</sub>, RQAOA<sub>1</sub>, and the best classical algorithm by Newman [14] for the MAX-3-CUT problem on these graphs. We observe that RQAOA<sub>1</sub> consistently outperforms QAOA<sub>1</sub> and that for certain regimes  $(n, d)$ , RQAOA<sub>1</sub> even outperforms Newman’s algorithm. However, for other families of graphs, Newman’s algorithm is capable of outputting near-optimal solutions, therefore outperforming RQAOA<sub>1</sub>.

## A.1.2 Individual Contribution

I am the principal author of this article. The idea for this work came after Article III) [3] in this thesis, where we introduced RQAOA to circumvent the limitations of QAOA that were found in this article. I was in charge of the numerical methods used in this article, with the exception of the angle optimization outlined in Appendix A, and was significantly involved in proving the main result (Theorem 3.1.) and the write-up of the article, with the exception of Section 4.

# Permission to include:

Sergey Bravyi, Alexander Kliesch, Robert Koenig, Eugene Tang  
“Hybrid quantum-classical algorithms for approximate graph coloring”  
*Quantum* 6, 678 (2022)

← → ↻ 🏠 <https://quantum-journal.org/about/terms-and-conditions/#publication> 200% ☆ ☰



## Copyright of works published by Quantum

All manuscripts and other parts of works that were previously submitted to Quantum and then published by Quantum, as well as the associated meta-data, including for example a work’s title, abstract, author list, figures, datasets, or popular summary, are published under the [Creative Commons Attribution 4.0 International \(CC BY 4.0\)](#) licence.

For material associated with a manuscript, such as that linked to from the manuscript, or a work’s page on Quantum’s website, especially if hosted on other platforms, other licences can apply.

Each owner of copyright on parts or the entirety of a work submitted to or published by Quantum retains their copyright.

# Hybrid quantum-classical algorithms for approximate graph coloring

Sergey Bravyi<sup>1</sup>, Alexander Kliesch<sup>2</sup>, Robert Koenig<sup>3</sup>, and Eugene Tang<sup>4</sup>

<sup>1</sup>IBM Quantum, IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA

<sup>2</sup>Zentrum Mathematik, Technical University of Munich, 85748 Garching, Germany

<sup>3</sup>Institute for Advanced Study & Zentrum Mathematik, Technical University of Munich, 85748 Garching, Germany

<sup>4</sup>Institute for Quantum Information and Matter, Caltech, Pasadena, CA 91125

We show how to apply the recursive quantum approximate optimization algorithm (RQAOA) to MAX- $k$ -CUT, the problem of finding an approximate vertex  $k$ -coloring of a graph. We compare this proposal to the best known classical and hybrid classical-quantum algorithms. First, we show that the standard (non-recursive) QAOA fails to solve this optimization problem for most regular bipartite graphs at any constant level  $p$ : the approximation ratio achieved by QAOA is hardly better than assigning colors to vertices at random. Second, we construct an efficient classical simulation algorithm which simulates level-1 QAOA and level-1 RQAOA for arbitrary graphs. In particular, these hybrid algorithms give rise to efficient classical algorithms, and no benefit arising from the use of quantum mechanics is to be expected. Nevertheless, they provide a suitable testbed for assessing the potential benefit of hybrid algorithm: We use the simulation algorithm to perform large-scale simulation of level-1 QAOA and RQAOA with up to 300 qutrits applied to ensembles of randomly generated 3-colorable constant-degree graphs. We find that level-1 RQAOA is surprisingly competitive: for the ensembles considered, its approximation ratios are often higher than those achieved by the best known generic classical algorithm based on rounding an SDP relaxation. This suggests the intriguing possibility that higher-level RQAOA may be a potentially useful algorithm for NISQ devices.

## 1 Introduction

Combinatorial optimization is currently considered one of the most promising areas of application of near-term quantum devices. One of the primary restrictions of such devices is the fact that they may only execute short-depth circuits with reasonable fidelity. This is a consequence of the lack of sophisticated fault-tolerance

mechanisms. Variational quantum algorithms such as the quantum approximate optimization algorithm (QAOA) [7] can deal with this restriction because parameters such as the circuit depth/architecture can be chosen according to existing experimental restrictions. This is in contrast to more involved quantum algorithms e.g., for algebraic problems: these may involve circuit depths scaling with the problem size and have additional requirements on the connectivity of available inter-qubit operations.

QAOA and similar proposals are hybrid algorithms: Here we envision operations on the quantum device to be supplemented by efficient, that is, polynomial-time classical processing. As a result, the algorithmic capabilities of such a hybrid setup should be compared to the class of efficient (polynomial-time) classical probabilistic algorithms. This means that corresponding proposals face stiff competition against decades of algorithms research in classical computer science. While there are complexity-theoretic arguments underscoring the power of e.g., constant-depth quantum circuits, the jury is still out on whether hybrid algorithms for near-term devices can indeed provide a provable computational advantage over comparable classical algorithms.

Recent no-go results show limitations of variational quantum algorithms for the well-studied MAX-CUT problem: in [3], we showed that the Goemans-Williamson-algorithm – the best known classical algorithm for this problem – outperforms QAOA for any constant level  $p$  (which amounts to constant-depth for bounded-degree graphs) in terms of the achieved approximation ratio. This result extends to more general, possibly non-uniform  $\mathbb{Z}_2$ -symmetric hybrid local quantum algorithms, and shows that corresponding circuits will need a circuit depth growing at least logarithmically with the problem size to yield a better approximation ratio. More recently, Farhi, Gamarnik and Gutmann [6] exploited the spatial uniformity of the QAOA algorithm to give an extremely elegant argument demonstrating a similar logarithmic-depth lower bound for QAOA for random  $d$ -regular graphs. The title of reference [6] aptly summarizes this conclusion using the words “the QAOA needs to see the whole graph”.

To go beyond these negative results, one could attempt to simply use logarithmic-depth circuits with the reasoning that for small to intermediate problem sizes, the corresponding circuit depths may still be amenable to realization by a near-term device. In addition to the problem of fault-tolerance, the potential merits of this idea are unfortunately difficult to assess by means of classical simulation. Indeed, the very idea that this approach yields computational benefits over classical algorithms is in tension with efficient classical simulability of corresponding quantum processes.

An alternative to this necessarily limited approach is to try to find new ways of leveraging the information-processing capabilities of short-depth circuits by introduction of non-local (classical) pre- and post-processing steps, thereby sidestepping the key assumption of locality in the aforementioned no-go results. As long as one restricts to procedures where these additional processing steps can be executed efficiently (e.g., in polynomial time) and the quantum device is used only a polynomial number of times, the resulting hybrid algorithm still has the feature of being efficiently executable with near-term hardware. At the same time, it may provide additional descriptive power, ultimately (hopefully) resulting in improved approxi-

mation ratios.

The recursive quantum approximate optimization algorithm (RQAOA) is a proposal of this kind which makes use of QAOA (with a constant level  $p$ ) in a recursive fashion with the goal of improving approximation ratios. Numerical evidence obtained for MAX-CUT on random graphs indicates that even level-1 RQAOA significantly improves upon QAOA (at the same level) for the MAX-CUT problem [3]. While the problem of establishing a rigorous lower bound on the approximation ratio achieved by this hybrid algorithm remains open (except for very special families of instances), this indicates that RQAOA – especially at higher levels  $p$  – has the potential to yield results competitive with the best known classical algorithms.

Here we consider the MAX- $k$ -CUT problem, an optimization version of the graph  $k$ -coloring problem. Suppose  $G = (V, E)$  is a graph with  $n = |V|$  vertices and  $e = |E|$  edges. Given an integer  $k \geq 2$ , the goal is to find an approximate  $k$ -coloring of vertices of  $G$  which maximizes the number of edges whose endpoints have different colors. For each vertex  $j \in V$ , introduce a variable  $x_j \in \mathbb{Z}_k$  which represents a color assigned to  $j$ . The  $k$ -coloring cost function to be maximized is defined as

$$C(x) = \sum_{(i,j) \in E} (1 - \delta_{x_i, x_j}) \quad \text{for} \quad x \in \mathbb{Z}_k^n. \quad (1)$$

The performance of a  $k$ -coloring algorithm on a given graph  $G$  is usually quantified by its approximation ratio  $\alpha$ , that is, the ratio between the expected value of the cost function  $C(x)$  on a coloring  $x$  produced by the algorithm and the maximum value  $\max_x C(x)$ .

The MAX- $k$ -CUT problem can also be viewed as an anti-ferromagnetic  $k$ -state Potts model. The standard MAX-CUT problem corresponds to  $k = 2$ . The problem is well-studied. Consider first the special case where  $G$  is a  $k$ -colorable graph. Clearly, a uniformly random assignment of colors  $x$  achieves an approximation ratio of  $1 - 1/k$  on average. For the case where  $k$  is a power of two, Cho, Rajee and Sarrafzadeh [11] constructed an  $O((e + n) \log k)$ -time algorithm which achieves an approximation ratio of  $1 - 1/k(1 - 1/n)^{\log k}$ , improving upon a deterministic  $O(enk)$ -time algorithm [21] achieving the same ratio  $1 - 1/k$  as random coloring (and obtained by derandomizing the latter). In the general case, when  $G$  may not be  $k$ -colorable, Frieze and Jerrum [8] gave an algorithm achieving an approximation ratio  $1 - 1/k + 2 \log k/k^2$  for arbitrary sufficiently large  $k$ . This is known to be close to optimal since no polynomial-time algorithm can achieve an approximation ratio better than  $1 - 1/(34k)$  unless  $P = NP$  [12], and is indeed optimal if one assumes the Unique Games Conjecture [13]. Frieze and Jerrum’s algorithm is based on an SDP relaxation and a randomized rounding scheme inspired by Goemans and Williamson’s algorithm for MAX-CUT [10] and comes with detailed estimates of the approximation ratio  $\alpha_k$  achieved by the algorithm, namely

$$\begin{aligned} \alpha_2 &\geq 0.878567 \\ \alpha_3 &\geq 0.800217 \\ \alpha_4 &\geq 0.850304 \\ \alpha_5 &\geq 0.874243 \end{aligned}$$

Here the bound on  $\alpha_2$  matches the Goemans-Williamson algorithm for MAX-CUT. This was further improved by a new algorithm by Klerk et al. [14] with a guaranteed approximation ratio of

$$\begin{aligned}\alpha_3 &\geq 0.836008 \\ \alpha_4 &\geq 0.857487 .\end{aligned}\tag{2}$$

Their algorithm is based on the properties of the Lovasz  $\vartheta$ -function and achieves the best currently known approximation ratio (for polynomial-time algorithms) for  $k = 3$ . Goemans and Williamson themselves [9] independently introduced an algorithm based on so-called “complex semidefinite programming” that matches this ratio for  $k = 3$ . Unfortunately, the analysis involved in establishing the bound (2) is significantly more complicated than the MAX-CUT case and is not known to be generalizable to arbitrary  $k$ . Fortunately, Newman [17] describes a simple rounding procedure that leads to an algorithm provably matching the bound (2) while only being slightly worse than Frieze/Jerrum for larger values of  $k$ . Finally, it should also be noted that dense 3-colorable graphs can be 3-colored in (randomized) polynomial time [1].

Here we study approximation ratios achieved by QAOA and RQAOA, respectively for MAX- $k$ -CUT with  $k > 2$ , and compare these to those achieved by the best known classical approximation algorithms for this problem discussed above.

**Outline** In Section 2, we discuss how to formulate QAOA and RQAOA for the MAX- $k$ -CUT-problem. In Section 3, we establish limitations on constant-level QAOA. In Section 4, we describe an efficient classical simulation algorithm for simulating level-1 QAOA and level-1 RQAOA. In Section 5, we discuss our numerical findings comparing level-1 QAOA and level-1 RQAOA with the best known classical algorithm for MAX- $k$ -CUT.

## 2 Hybrid algorithms for MAX- $k$ -CUT

Here we give a brief description of how QAOA and RQAOA can be adapted to apply to MAX- $k$ -CUT for any  $k \geq 2$ .

### 2.1 QAOA to MAX- $k$ -CUT

Solving MAX- $k$ -CUT by level- $p$  QAOA (in the following denoted by QAOA $_p$ ) proceeds in an established fashion as in the case of MAX-CUT (i.e.,  $k = 2$ ). One notable feature is that in the MAX- $k$ -CUT problem, it is natural to work with  $n$  qudits of dimension  $k$  each instead of qubits (all quantum circuits considered below can be simulated on a standard qubit-based quantum computer with a constant-factor overhead). We use the  $n$ -qudit cost function Hamiltonian

$$C = \sum_{1 \leq i < j \leq n} \sum_{b \in \mathbb{Z}_k} J_{i,j}(b) \Pi_{i,j}(b)\tag{3}$$

where  $J_{i,j}(b)$  are real coefficients and

$$\Pi(b) = \sum_{a \in \mathbb{Z}_k} |a, a+b\rangle \langle a, a+b|$$

is a diagonal projector acting on  $\mathbb{C}^k \otimes \mathbb{C}^k$ . Here and below the addition of color indices is performed modulo  $k$ . The subscripts  $i, j$  in  $\Pi_{i,j}(b)$  indicate which pair of qudits is acted upon by  $\Pi(b)$ . By definition,  $\Pi_{i,j}(b) = \Pi_{j,i}(-b)$ .

Equation (3) defines a general class of cost function Hamiltonians. The MAX- $k$ -CUT problem on a graph  $G = (V, E)$  associated with the cost function (1) corresponds to the choice of coefficients

$$J_{i,j}(b) = \begin{cases} 0, & (i, j) \notin E \\ 1 - \delta_{b,0}, & (i, j) \in E. \end{cases} \quad (4)$$

The Hamiltonian  $C$  defined Eq. (3) commutes with the symmetry operator  $X^{\otimes n}$ , where

$$X = \sum_{c \in \mathbb{Z}_k} |c+1\rangle \langle c|$$

is the generalized Pauli- $X$  operator. This is analogous to the  $\mathbb{Z}_2$ -symmetry of the Ising model exploited in [3]. Motivated by the corresponding ansatz for MAX-CUT, we generalize the level- $p$  QAOA ansatz to qudits as

$$|\psi(\beta, \gamma)\rangle = U(\beta, \gamma)|+\rangle^n, \quad (5)$$

where the level- $p$  QAOA unitary is given by

$$U(\beta, \gamma) = \prod_{t=1}^p B(\beta^{(t)})^{\otimes n} e^{-i\gamma^{(t)}C},$$

with  $\beta = (\beta^{(1)}, \dots, \beta^{(p)}) \in (\mathbb{R}^k)^p$ ,  $\gamma = (\gamma^{(1)}, \dots, \gamma^{(p)}) \in \mathbb{R}^p$  and where for  $\beta \in \mathbb{R}^k$ , the unitary  $B(\beta) : \mathbb{C}^k \rightarrow \mathbb{C}^k$  is diagonal in the eigenbasis of  $X$  and given by

$$B(\beta) = \sum_{a \in \mathbb{Z}_k} e^{i\beta_a} |\phi_a\rangle \langle \phi_a|, \quad |\phi_a\rangle \equiv Z^a|+\rangle. \quad (6)$$

In this expression,

$$Z = \sum_{a \in \mathbb{Z}_k} \omega^a |a\rangle \langle a|, \quad \omega \equiv e^{2\pi i/k}$$

is the generalized Pauli- $Z$  operator, whereas  $|+\rangle \in \mathbb{C}^k$  is the  $+1$  eigenvector of  $X$ , that is,  $|+\rangle = k^{-1/2} \sum_{b \in \mathbb{Z}_k} |b\rangle$ . Level- $p$  QAOA for MAX- $k$ -CUT proceeds by

- (i) first maximizing the expected value  $\langle \psi|C|\psi\rangle$  where  $|\psi\rangle \equiv |\psi(\beta, \gamma)\rangle$  over  $(\beta, \gamma) \in (\mathbb{R}^k)^p \times \mathbb{R}^p$ .
- (ii) then measuring an energy-maximizing state  $|\psi_*\rangle = |\psi(\beta_*, \gamma_*)\rangle$  in the computational basis.

The output of this process is a coloring  $x \in \mathbb{Z}_k^n$  achieving an expected approximation ratio

$$\max_{\beta, \gamma} \langle \psi|C|\psi\rangle / \max_x C(x).$$

This concludes the description of QAOA $_p$ .



## 2.2 Adapting RQAOA to MAX- $k$ -CUT

The RQAOA proceeds by successively reducing the size of the problem, eliminating a single variable in each step by a procedure called correlation rounding. To formulate RQAOA for MAX- $k$ -CUT, we begin by noting that the family of Hamiltonians defined by (3) is closed under variable eliminations (up to irrelevant additive constants).

Level- $p$  RQAOA proceeds by iterative applications of several single variable elimination steps as described by (i) and (ii) described below. We note that in the very first variable elimination step, the scalar  $M_{i,j}(b)$  (for a fixed pair of vertices  $(i, j)$ ) computed in (i) depends only on whether or not  $b$  is non-zero. This is a consequence of the specific form (4) of the coefficients  $J_{i,j}(b)$  in the (initial) MAX- $k$ -CUT cost function Hamiltonian  $C$  (Eq. (3)). However, this is no longer necessarily the case after one or more variable elimination steps have been completed since the cost function Hamiltonian is updated according to (ii).

A single variable elimination step of level- $p$  RQAOA works as follows:

- (i) First, maximize the expected value  $\langle \psi | C | \psi \rangle$  with  $|\psi\rangle = |\psi(\beta, \gamma)\rangle$  over  $(\beta, \gamma) \in (\mathbb{R}^k)^p \times \mathbb{R}^p$ . Then compute the mean value

$$M_{i,j}(b) = \langle \psi | \Pi_{i,j}(b) | \psi \rangle \quad \text{for all pairs of vertices } (i, j) .$$

Note that  $0 \leq M_{i,j}(b) \leq 1$  since  $\Pi_{i,j}(b)$  is a projector.

- (ii) Next, find a pair of vertices  $(i, j)$  and a color  $b \in \mathbb{Z}_k$  with the largest magnitude of  $M_{i,j}(b)$  (breaking ties arbitrarily). Then impose the constraint

$$x_j = x_i + b \pmod{k} , \tag{7}$$

restricting the search space to the span of computational basis vectors  $|x\rangle$  associated with colorings  $x \in (\mathbb{Z}_k)^n$  satisfying (7). Observe that  $|\psi\rangle$  has support on such basis states if and only if  $M_{i,j}(b) = 1$ . To eliminate the variable  $x_i$ , the constraint (7) is inserted into the cost function Hamiltonian as follows: use the identity

$$\Pi_{i,j}(b)\Pi_{j,h}(a-b) = \Pi_{i,j}(b)\Pi_{i,h}(a)$$

which holds for all  $h \notin \{i, j\}$  and all  $a \in \mathbb{Z}_k$ . Thus  $\Pi_{i,h}(a) = \Pi_{j,h}(a-b)$  on the subspace satisfying the constraint. Replacing  $\Pi_{i,h}(a)$  by  $\Pi_{j,h}(a-b)$  in the cost function Hamiltonian for all  $h \notin \{i, j\}$  one gets a new Hamiltonian  $C'$  of the form (3) (up to an additive constant) acting on  $n-1$  variables, i.e.,  $C'$  acts trivially on the  $i$ -th qudit. The cost function  $C'$  is defined on a graph  $G'$  with  $n-1$  vertices obtained from  $G$  by identifying the vertices  $i$  and  $j$ .

By construction, the maximum energy of  $C'$  coincides with the maximum energy of  $C$  over the subset of assignments satisfying the constraints (7). Since the new Hamiltonian  $C'$  acts trivially on the qudit  $i$ , this qudit can be removed from the simulation. This completes the variable elimination step.

RQAOA $_p$  executes several variable elimination steps in succession, eliminating one variable in each recursion until the number of variables reaches a predefined

cutoff value  $n_c$ . The remaining Hamiltonian then only depends on  $n_c$  variables is minimized by a purely classical algorithm, for example brute-force search. An (approximate) solution  $x \in \mathbb{Z}_k^n$  of the original problem can then be obtained recursively by reconstructing eliminated variables using the constraints (7).

Since the introduction of RQAOA in [3], other modifications of QAOA involving iterated rounding procedures have been proposed, see e.g., [16, Section V.A]. In contrast to the variant discussed there, RQAOA's rounding procedure is deterministic and relies on rounding correlations between qudits rather than individual spin polarizations.

Further variations of RQAOA have been proposed and studied in [5, 20]. There, the authors consider “warm-starting” the algorithm by beginning with a solution returned by an efficient classical algorithm (for the case of MAX-CUT, the Goemans-Williamson algorithm) instead of the standard product state. They provided numerical evidence that both QAOA and RQAOA can achieve better performance when supplemented with “warm-starting”, making this a promising potential avenue for future investigations.

### 3 Limitations of QAOA<sub>p</sub> applied to MAX- $k$ -CUT

Limitations for level- $p$  QAOA with  $p = O(\log n)$  applied to MAX-CUT were obtained for ensembles of random  $d$ -regular graphs in [6]. Here we show analogous results for QAOA applied to MAX- $k$ -CUT for  $k > 2$ . Our main result is the following theorem. It shows that unless the level  $p$  of QAOA grows at least logarithmically with  $n$ , QAOA cannot be more than marginally better (for large  $n$  and  $d$ ) than randomly guessing a coloring, for most  $d$ -regular bipartite graphs. We denote by  $\mathcal{G}_{n,d}^{bi}$  the ensemble of uniformly random  $d$ -regular bipartite graphs on  $n$  vertices. For a graph  $G$ , we denote by  $\text{MC}_k(G) = \max_{x \in \mathbb{Z}_k^n} C(x)$  the maximum  $k$ -cut of  $G$ . Note that  $\text{MC}_k(G) = nd/2$  for any  $d$ -regular bipartite graph  $G$ . Let  $\alpha_p^{\text{MC}_k}(G)$  be the approximation ratio of MAX- $k$ -CUT QAOA<sub>p</sub> applied to  $G$ , that is,  $\alpha_p^{\text{MC}_k}(G) = \langle \psi(\theta_*) | C(G) | \psi(\theta_*) \rangle / \text{MC}_k(G)$  where  $\theta_* = (\beta_*, \gamma_*) \in (\mathbb{R}^k)^p \times \mathbb{R}^p$  is an optimal set of angles, and where  $C(G)$  is the cost function Hamiltonian for  $G$  defined by Eq. (3).

**Theorem 3.1.** *There is a constant  $\zeta > 0$  such that*

$$\Pr_{G \sim \mathcal{G}_{n,d}^{bi}} \left[ \alpha_p^{\text{MC}_k}(G) \geq (1 - 1/k) + o_d(1) + o_n(1) \right] \leq o(1)$$

for all degrees  $d$  satisfying  $d \geq \zeta$  and  $d = o(\sqrt{n})$  and all levels  $p < \frac{1}{2} \log_d n$ .

Our approach generally follows the basic idea of [6] and also exploits the fact that QAOA is a local, and furthermore uniform algorithm. In contrast to the main result of [6], which is expressed as an upper bound on the expected approximation ratio (over the choice of graph) achieved by QAOA, Theorem 3.1 shows that the approximation ratio is upper bounded for almost all graphs in the considered ensemble. Analogous to the setting of [6], where it is shown that the approximation ratio converges to  $1/2$  in the limit of large  $d$ , we show that the limiting value is

the approximation ratio achieved by random guessing. In the analysis of [6], the explicitly known expected maximal cut size for random  $d$ -regular graphs is used. The analogous value for MAX- $k$ -CUT is not known for random  $d$ -regular graphs. Instead, we rely on an upper bound on the size of the maximum  $k$ -cut for typical  $d$ -regular random graphs obtained in the analysis of an SDP-relaxation from [4]. In our analysis, we pay special attention to the graph-dependence of the optimal QAOA angles when computing QAOA approximation ratios: here we make use of the fact that the figure of merit (the average energy) for different typical graphs differs only by a negligible amount for any fixed angles.

Given a graph  $G = (V, E)$ , let  $(i, j) = e \in E$  be an edge in  $G$ . We write  $N_p(e)$  to denote the  $p$ -local neighborhood of  $e$ , i.e., the subgraph induced by the set of all vertices  $h$  such that  $d(h, i) \leq p$  or  $d(h, j) \leq p$ . Let  $T_{p,d}$  denote the tree such that all vertices except the leaves are  $d$ -regular and such that there exists an edge  $(i, j)$  such that all leaves are distance  $p$  from either  $i$  or  $j$ . Let  $n_{T_{p,d}}(G)$  denote the number of edges  $e$  in a graph  $G$  such that  $N_p(e) \not\cong T_{p,d}$ . The key insight of [6] is the fact that for  $p$  sufficiently small compared to  $n$ , it holds with high probability that almost all  $p$ -local neighborhoods of a random  $d$ -regular graph look like  $T_{p,d}$ . This is expressed by the following lemma, where we denote by  $\mathcal{G}_{n,d}$  the uniform distribution over  $d$ -regular graphs on  $n$  vertices.

**Lemma 3.2.** [6] *Let  $n, d$  and  $A \in (0, 1)$  be given. Suppose that*

$$(d - 1)^{2p} < n^A . \quad (8)$$

*Then there exists some constant  $A' \in (A, 1)$  such that*

$$\mathbb{E}_{G \sim \mathcal{G}_{n,d}} [n_{T_{p,d}}(G)] = O(n^{A'}) .$$

*In particular, by Markov's inequality,*

$$\Pr_{G \sim \mathcal{G}_{n,d}} [n_{T_{p,d}}(G) \geq n^B] = O(n^{A'-B}) = o(1) \quad \text{for all } B \in (A', 1) . \quad (9)$$

*Moreover, the same result also holds for the ensemble  $\mathcal{G}_{n,d}^{bi}$  of random  $d$ -regular bipartite graphs.*

The proof of Lemma 3.2 relies on the fact that the number of cycles of length  $\ell$  in a random  $d$ -regular graph is Poisson distributed, allowing an upper bound to be established on the expected number of cycles below a certain length [23]. This latter bound applies to both random  $d$ -regular graphs as well as random bipartite  $d$ -regular graphs. Both ensembles will be of relevance in the proof of Theorem 3.1.

Let us now fix some  $B \in (A', 1)$ . We will say that a graph  $G$  on  $n$  vertices is  $T$ -typical if  $n_{T_{p,d}}(G) < n^B$ . Lemma 3.2 says that a random (possibly bipartite)  $d$ -regular graph  $G$  is  $T$ -typical with high probability. Since QAOA is a local, and moreover uniform, algorithm, the performance of QAOA $_p$  on any given graph can be related to the performance of QAOA $_p$  on the induced  $p$ -neighborhood of each edge. Lemma 3.2 therefore suggests that to study QAOA $_p$  for generic  $d$ -regular graphs, it suffices to consider the behavior of QAOA $_p$  on  $T_{p,d}$ .

Let now  $C(G)$  be the cost function Hamiltonian for the MAX- $k$ -CUT problem associated with  $G$  (see Eq. (3)), and let  $\psi(\theta)$  denote the level- $p$  QAOA state defined

by Eq. (5), where we use the shorthand  $\theta = (\beta, \gamma)$  for the collection of all angles. Then the expectation value for MAX- $k$ -CUT QAOA $_p$  on  $G = (V, E)$  can be written as

$$\langle \psi(\theta) | C(G) | \psi(\theta) \rangle = \sum_{(i,j) \in E} \langle \psi(\theta) | C_{i,j} | \psi(\theta) \rangle ,$$

where  $C_{i,j}$  is the term in the Hamiltonian (3) acting non-trivially on both qudits  $i$  and  $j$ . Note that due to the locality of the QAOA unitary, each of the individual terms  $\langle \psi(\theta) | C_{i,j} | \psi(\theta) \rangle$  in this expression is a function of the subgraph  $N_p(ij)$  and  $\theta$ .

Now, let  $G$  be a  $d$ -regular  $T$ -typical graph. Each of the local terms  $\langle \psi(\theta) | C_{i,j} | \psi(\theta) \rangle$  is bounded above by 1 since  $C_{i,j}$  is a projection. Since for a  $T$ -typical graph  $G$ , all but  $n_{T_{p,d}}(G) < n^B$  of the edges satisfy  $N_p(e) \cong T_{p,d}$ , it follows that

$$\begin{aligned} \langle \psi(\theta) | C(G) | \psi(\theta) \rangle &= \left( \frac{nd}{2} - n_{T_{p,d}}(G) \right) C_T(\theta) + \sum_{(i,j): N_p(ij) \not\cong T_{p,d}} \langle \psi(\theta) | C_{i,j} | \psi(\theta) \rangle \\ &= \frac{nd}{2} C_T(\theta) + O(n^B) . \end{aligned} \quad (10)$$

In this expression, the function  $C_T(\theta) = \langle \psi(\theta) | C_{i',j'} | \psi(\theta) \rangle$  is the local term associated with any one edge  $(i', j')$  such that  $N_p(i'j') \cong T_{p,d}$ . Importantly, the function  $C_T(\cdot)$  depends on  $(p, d)$  only, i.e., it is universal for all  $T$ -typical graphs  $G$ .

An important consequence is that the maximal expectation value achieved by level- $p$  QAOA is roughly identical for all  $T$ -typical graphs  $G$ : if  $G_1$  and  $G_2$  are  $T$ -typical, then

$$\left| \max_{\theta} \langle \psi(\theta) | C(G_1) | \psi(\theta) \rangle - \max_{\theta} \langle \psi(\theta) | C(G_2) | \psi(\theta) \rangle \right| = O(n^B) . \quad (11)$$

This follows from the fact that

$$\left| \langle \psi(\theta) | C(G_1) | \psi(\theta) \rangle - \langle \psi(\theta) | C(G_2) | \psi(\theta) \rangle \right| = O(n^B) \quad \text{for any set of angles } \theta \quad (12)$$

as a consequence of (10), and the inequality  $|\|f\|_{\infty} - \|g\|_{\infty}| \leq \|f - g\|_{\infty}$ . Specializing Eq. (12) to  $\theta = \arg \max_{\theta} \langle \psi(\theta) | C(G_1) | \psi(\theta) \rangle$  also shows that a choice of optimal angles for any given  $T$ -typical graph  $G_1$  will also be nearly optimal for all other  $T$ -typical graphs  $G_2$ .

We are going to establish an upper bound on the performance of QAOA on  $T$ -typical graphs. To this end, we use the following upper bound on the typical size of the maximum  $k$ -cut of a graph  $G$  in the ensemble  $\mathcal{G}_{n,d}$  of  $d$ -regular graphs. Then the following holds:

**Theorem 3.3.** [4, Theorem 19] *There exists constants  $\lambda, \zeta > 0$  such that*

$$\Pr_{G \sim \mathcal{G}_{n,d}} \left[ \text{MC}_k(G) \leq \left( 1 - \frac{1}{k} \right) \frac{nd}{2} + \lambda n \sqrt{d} \right] \geq 1 - e^{-2n}$$

for all degrees  $d$  satisfying  $d \geq \zeta$  and  $d = o(\sqrt{n})$ .

This implies that for all degrees  $d$  satisfying the hypotheses of Theorem 3.3, we have

$$\mathbb{E}_{G \sim \mathcal{G}_{n,d}}[\text{MC}_k(G)] \leq \left(1 - \frac{1}{k}\right) \frac{nd}{2} + \lambda n \sqrt{d} + o(n). \quad (13)$$

**Lemma 3.4.** *Let  $n, d$  satisfy the hypotheses of Theorem 3.3 and let  $p, A$  be such that Eq. (8) holds. Then there exists a constant  $B' \in (0, 1)$  such that the following holds for sufficiently large  $n$ : There exists a  $T$ -typical  $d$ -regular graph  $G$  satisfying*

$$\text{MC}_k(G) < \mathbb{E}_{G \sim \mathcal{G}_{n,d}}[\text{MC}_k(G)] + n^{B'}. \quad (14)$$

*Proof.* Let  $\mathcal{T}$  denote the set of  $T$ -typical graphs, and let us abbreviate  $E = \mathbb{E}_{G \sim \mathcal{G}_{n,d}}[\text{MC}_k(G)]$ . We show that a randomly chosen  $d$ -regular graph  $G \sim \mathcal{G}_{n,d}$  satisfies the desired properties (i.e.,  $G \in \mathcal{T}$  and Eq. (14)) with non-zero probability. Indeed, we have by the union bound

$$\Pr_{G \sim \mathcal{G}_{n,d}} \left[ \mathcal{G} \notin \mathcal{T} \text{ or } \text{MC}_k(G) \geq E + n^{B'} \right] \leq \Pr_{G \sim \mathcal{G}_{n,d}} [G \notin \mathcal{T}] + \Pr_{G \sim \mathcal{G}_{n,d}} \left[ \text{MC}_k(G) \geq E + n^{B'} \right]. \quad (15)$$

We show that sum on the rhs is strictly less than 1 for a suitable choice of  $B' \in (0, 1)$  in the limit  $n \rightarrow \infty$ , implying the claim. Indeed, by Eq. (9) of Lemma 3.2, there is some  $A' \in (A, 1)$  such that

$$\Pr_{G \sim \mathcal{G}_{n,d}} [G \notin \mathcal{T}] = O(n^{A'-B}) \quad \text{whenever} \quad B \in (A', 1). \quad (16)$$

On the other hand, we have with Markov's inequality

$$\Pr_{G \sim \mathcal{G}_{n,d}} \left[ \text{MC}_k(G) \geq E + n^{B'} \right] = \frac{1}{1 + n^{B'}/E} \leq 1 - \frac{1}{2} n^{B'}/E$$

by Taylor series expansion. Note that the latter is well-defined, i.e.,  $n^{B'}/E \rightarrow 0$  for  $n \rightarrow \infty$  for all  $B' \in (0, 1)$ , since we have  $E = \Omega(n)$ . This is because a max  $k$ -cut is at least as large as a max 2-cut, and the expected max 2-cut scales as  $\Theta(n)$ . Inserting (13), we conclude that

$$\Pr_{G \sim \mathcal{G}_{n,d}} \left[ \text{MC}_k(G) \geq E + n^{B'} \right] \leq 1 - \frac{1}{2} \left[ \left( \left(1 - \frac{1}{k}\right) \frac{d}{2} + \lambda \sqrt{d} \right)^{-1} n^{B'-1} + o(n^{B'-1}) \right]. \quad (17)$$

Combining Eqs. (15), (16), (17) and comparing exponents of  $n$ , we conclude that

$$\Pr_{G \sim \mathcal{G}_{n,d}} \left[ \mathcal{G} \notin \mathcal{T} \text{ or } \text{MC}_k(G) \geq E + n^{B'} \right] < 1 \quad \text{for sufficiently large } n$$

$$\text{whenever } A' - B < B' - 1.$$

The Lemma therefore holds for any choice of  $B' \in (1 - B + A', 1)$ .  $\square$

Now, let us fix a  $T$ -typical graph  $G_*$  from Lemma 3.4 with some chosen constant  $B'$ . Then we have

$$\max_{\theta} \langle \psi(\theta) | C(G_*) | \psi(\theta) \rangle \leq \text{MC}_k(G_*) < \mathbb{E}_{G \sim \mathcal{G}_{n,d}} [\text{MC}_k(G)] + n^{B'}.$$

The first inequality follows from the fact that the expected value of the MAX- $k$ -CUT cost function for colorings returned by QAOA cannot exceed the actual maximum. From inequality (13), we then have the bound

$$\max_{\theta} \langle \psi(\theta) | C(G_*) | \psi(\theta) \rangle \leq \left( \left(1 - \frac{1}{k}\right) \frac{d}{2} + \lambda \sqrt{d} \right) \cdot n + o(n). \quad (18)$$

From the uniform bound Eq. (11) between the maxima of  $T$ -typical graphs, the same bound holds for any  $T$ -typical graph  $G$  with the same  $n$  and  $d$  as  $G_*$ , that is

$$\left| \max_{\theta} \langle \psi(\theta) | C(G) | \psi(\theta) \rangle - \max_{\theta} \langle \psi(\theta) | C(G_*) | \psi(\theta) \rangle \right| = o(n).$$

To obtain a bound on the approximation ratio, we can focus our attention on  $d$ -regular bipartite graphs, which are guaranteed to have a maximum  $k$ -cut of size  $nd/2$  for any  $k$ . Since Lemma 3.2 applies equally to random bipartite graphs, so that a random bipartite  $d$ -regular graph is also  $T$ -typical with probability  $1 - o(1)$ . Letting  $G$  be a  $T$ -typical bipartite graph with  $n$  and  $d$  satisfying the hypothesis of Theorem 3.3, the bound (18) holds for  $G$ . Dividing through by  $nd/2$ , the approximation ratio  $\alpha_p^{\text{MC}_k}(G)$  is therefore bounded above by

$$\alpha_p^{\text{MC}_k}(G) = \frac{2}{nd} \max_{\theta} \langle \psi(\theta) | C(G) | \psi(\theta) \rangle \leq \left(1 - \frac{1}{k}\right) + o_d(1) + o_n(1).$$

This concludes the proof of Theorem 3.1.

## 4 Classical simulation of level-1 RQAOA

A polynomial-time classical algorithm for computing expectation values  $\langle \psi(\beta, \gamma) | Z_j Z_k | \psi(\beta, \gamma) \rangle$  for level-1 QAOA states  $|\psi(\beta, \gamma)\rangle$  and the MAX-CUT cost function was given by Wang et al. [22]. Our work [3] describes a generalized version of this algorithm applicable to any Ising-type cost function.

In this section, we consider the  $k$ -coloring cost function Hamiltonian (3) and more generally Hamiltonians of the form (3) and give a classical algorithm for computing expectation values  $\langle \psi | Z_u^r Z_v^s | \psi \rangle$ , where  $|\psi\rangle = |\psi(\beta, \gamma)\rangle$  is the level-1 QAOA state defined in Eq. (5) and  $Z_u$  is the generalized Pauli- $Z$  operator acting on a qudit  $u \in [n]$ . The algorithm has runtime  $O(k^5(d_u + d_v))$ , where  $d_j$  is the degree of a vertex  $j$ . For a constant number of colors  $k$ , this scales at most linearly with  $n$ . For constant-degree graphs, the computation requires a constant amount of time.

A natural application of our algorithm is finding optimal angles  $(\beta, \gamma)$  for level-1 QAOA states. Indeed, since the variational energy is a linear combination of the expected values  $\langle \psi | Z_u^r Z_v^s | \psi \rangle$ , it can be efficiently computed classically using our algorithm. This eliminates the need to prepare the variational state  $|\psi\rangle$  on

a quantum device for each intermediate choice of the angles  $(\beta, \gamma)$  in the energy optimization subroutine. In fact, the technique discussed here can easily be adapted to also efficiently compute partial derivatives  $\frac{\partial}{\partial \gamma} \langle \psi | Z_u^r Z_v^s | \psi \rangle$  and  $\frac{\partial}{\partial \beta_j} \langle \psi | Z_u^r Z_v^s | \psi \rangle$ , thus sidestepping the need of using a quantum device even if the latter is used to estimate the gradient of the cost function. However, once the optimal angles  $(\beta, \gamma)$  are found, a quantum device would be needed to prepare the optimal variational state and perform a measurement to obtain a classical solution of the problem. Since the final measurement is not used in the recursive version of QAOA, our algorithm enables efficient classical simulation of level-1 RQAOA in its entirety.

#### 4.1 General algorithm for level-1 QAOA-type expectation values

Our formulation extends beyond our QAOA Ansatz for MAX- $k$ -CUT and could be used in other contexts. Specifically, let  $C$  be a 2-local Hamiltonian on  $n$   $k$ -dimensional qudits given by

$$C = \sum_{1 \leq u < v \leq n} C_{u,v} ,$$

where we assume that the terms  $C_{u,v}$  and  $C_{u',v'}$  acting on different qudits  $u < v$  and  $u' < v'$  commute pairwise. We will also write  $C_{v,u} \equiv \text{SWAP} C_{u,v} \text{SWAP}^\dagger$  for the interaction term  $C_{u,v}$  acting on qudits  $(v, u)$  (in this order) when  $u < v$ . Let  $|+\rangle \in \mathbb{C}^k$  be a qudit state. We are interested in generalized QAOA states of the form

$$|\Psi(\beta, \gamma)\rangle = B(\beta)^{\otimes n} e^{i\gamma C} |+\rangle^{\otimes n} \quad (19)$$

where  $B(\beta) : \mathbb{C}^k \rightarrow \mathbb{C}^k$  is a unitary on  $\mathbb{C}^k$  for any  $\beta \in \mathbb{R}^k$ . We assume that  $B(-\beta) = B(\beta)^\dagger$  is the adjoint of  $B(\beta)$ . The states defined by Eq. (5) are a special case.

Let  $O$  be any two-qudit observable. Our goal is to compute the expectation value

$$\mu_{u,v}(O) = \langle \psi(\beta, \gamma) | O_{u,v} | \psi(\beta, \gamma) \rangle , \quad (20)$$

where the subscripts  $u, v$  indicate the pair of qudits acted upon by  $O$ . Define the density matrix

$$\rho_{u,v} = \text{Tr}_{w \notin \{u,v\}} \left( e^{i\gamma C} |+\rangle \langle +|^{\otimes n} e^{-i\gamma C} \right) .$$

One can compute  $\rho_{u,v}$  in time  $O(n)$  by initializing the pair of qudits  $u, v$  in the state  $e^{i\gamma H_{u,v}} |+\rangle^{\otimes 2}$  and sequentially coupling  $\{u, v\}$  to each of the remaining qudits  $w$  in the following way: qubit  $w$  is initialized in the state  $|+\rangle$  and then coupled to  $\{u, v\}$  by the respective terms in the cost function  $C$ , namely  $e^{i\gamma(C_{u,w} + C_{v,w})}$ . Finally,  $w$  is traced out.

The algorithm for computing  $\rho_{u,v}$  can be summarized as follows:

```

 $\eta \leftarrow e^{i\gamma C_{u,v}} |+\rangle\langle +|^{\otimes 2} e^{-i\gamma C_{u,v}}$ 
for  $w \in [n] \setminus \{u, v\}$  do
     $\eta \leftarrow \mathcal{E}_w(\eta)$ 
end for
return  $\eta$ 

```

where  $\mathcal{E}_w$  is a two-qudit quantum channel defined as

$$\mathcal{E}_w(\eta) = \text{Tr}_2 \left[ e^{i\gamma(C_{u,w} + C_{v,w})} (\eta \otimes |+\rangle\langle +|) e^{-i\gamma(C_{u,w} + C_{v,w})} \right]. \quad (21)$$

The final state  $\rho_{u,v}$  involves  $n - 2$  applications of  $\mathcal{E}_w$  in general. By restriction to  $w$  which interact non-trivially with  $\{u, v\}$ , this can be reduced to fewer than  $d_u + d_v$  applications, where  $d_u$  is the degree of  $u$  in the interaction graph associated with  $C$  (i.e.,  $(u, w)$  is an edge if and only if  $C_{u,w} \neq 0$ ). This improvement applies for example to the MAX- $k$ -CUT cost function Hamiltonian (3) for a bounded-degree graph.

Finally, the definition (19) of  $|\psi(\beta, \gamma)\rangle$  together with the assumed commutativity of the terms  $C_{u,v}$  in the Hamiltonian give that the expectation value (20) can be computed according to

$$\mu_{u,v}(O) = \text{Tr} \left( \rho_{u,v} B(-\beta)^{\otimes 2} O_{u,v} B(\beta)^{\otimes 2} \right). \quad (22)$$

This computation is illustrated in Fig. 1. To find the  $k$ -dependence of the overall complexity of this algorithm, we need to consider the evaluation of the superoperators  $\mathcal{E}_w$  defined by (21) and the expression (22) for the problem at hand.

## 4.2 Simulating QAOA<sub>1</sub> for MAX- $k$ -CUT

Here we specialize the algorithm from Section 4.1 to the MAX- $k$ -CUT problem. For this problem,  $B(\beta)$  is given by Eq. (6) whereas  $C$  (cf. Eq. (3)) is a diagonal 2-local Hamiltonian commuting with  $X^{\otimes n}$ . The most general diagonal 2-local Hamiltonian  $C$  commuting with  $X^{\otimes n}$  has the following form: Let  $\{h(a)\}_{a \in \mathbb{Z}_k}$  be a family of  $n \times n$ -matrices satisfying

$$\overline{h_{u,v}(r)} = h_{v,u}(r) = h_{u,v}(-r) \quad (23)$$

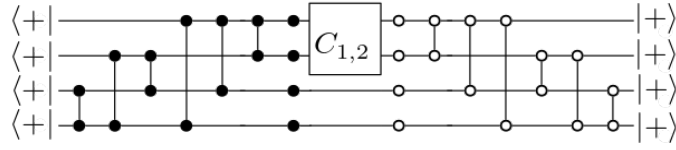
for all  $u, v \in [n]$  and  $r \in \mathbb{Z}_k$ . Then  $C$  can be written as

$$C = \sum_{1 \leq u < v \leq n} C_{u,v}, \quad C_{u,v} = \sum_{a \in \mathbb{Z}_k} h_{u,v}(a) Z_u^a Z_v^{-a} = \sum_{a \in \mathbb{Z}_k} h_{v,u}(a) Z_v^a Z_u^{-a}. \quad (24)$$

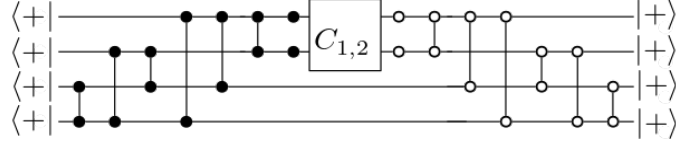
From Eq. (23) one gets  $C_{v,u} = C_{u,v}$  and  $C_{u,v}^\dagger = C_{u,v}$ . Let us agree that  $C_{u,u} = 0$  for all  $u$ . The Hamiltonian Eq. (3) takes the form (24) with  $h_{u,v}(a) = \frac{1}{k} \sum_{b \in \mathbb{Z}_k} J_{u,v}(b) \omega^{ab}$  since

$$\Pi_{u,v}(b) = \frac{1}{k} \sum_{a \in \mathbb{Z}_k} \omega^{ab} Z_u^a Z_v^{-a}.$$

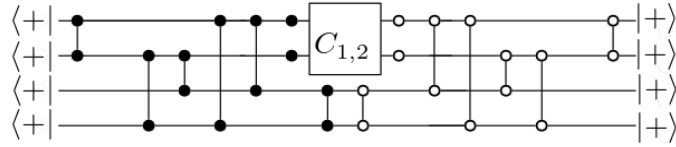




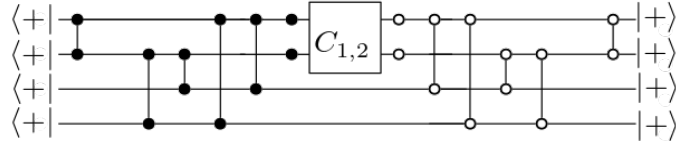
(a) The expression of interest.



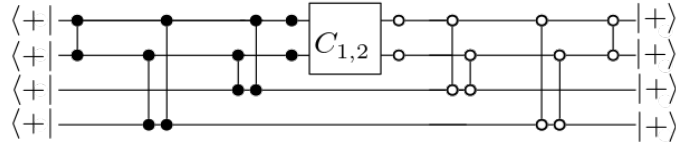
(b) Single-qubit unitaries  $B(\beta)$  not acting on the qudits  $\{1, 2\}$  cancel.



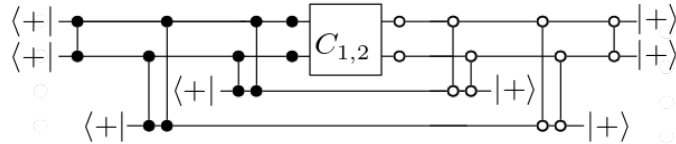
(c) The two unitaries  $e^{\pm i\gamma C_{1,2}}$  can be moved to the left and right. Unitaries  $e^{\pm i\gamma C_{p,q}}$  acting on qudits  $p, q \notin \{1, 2\}$  can be moved to the center.



(d) where they cancel with their adjoints.



(e) Again using commutativity of the operators  $C_{p,q}$ , the remaining unitaries can be grouped into pairs of the form  $e^{\pm i\gamma(C_{1,w}+C_{2,w})}$  where  $w \notin \{1, 2\}$ .



(f) The resulting expression can be interpreted as the result of applying a sequence of superoperators  $\mathcal{E}_w$  to the state  $e^{i\gamma C_{1,2}}|+\rangle\langle+|^{\otimes 2}e^{-i\gamma C_{1,2}}$ , and then taking the expectation of  $B(-\beta)^{\otimes 2}C_{1,2}B(\beta)^{\otimes 2}$ .

Figure 1: An expression of the form  $\langle\psi(\beta, \gamma)|C_{u,v}|\psi(\beta, \gamma)\rangle$ . Here we assumed that  $n = 4$  and  $(u, v) = (1, 2)$ . Single-qudit operators of the form  $B(\beta)$  are represented by circles, and their adjoints  $B(\beta)^\dagger$  by filled circles. Similarly, gates of the form  $e^{-i\gamma C_{p,q}}$  respectively  $e^{i\gamma C_{p,q}}$  for qudits  $p < q$  are represented by connecting lines between the corresponding qudits with empty respectively filled circles.

Note that

$$\langle a, b | e^{-i\gamma C_{p,q}} | a, b \rangle = \exp \left[ -i\gamma \hat{h}_{p,q}(a - b) \right] \quad (25)$$

where the Fourier transform  $\hat{h}(b)$  of  $h$  is defined as a hermitian  $n \times n$  matrix with entries

$$\hat{h}_{p,q}(b) = \sum_{a \in \mathbb{Z}_k} h_{p,q}(a) \omega^{ab}, \quad \omega \equiv e^{2\pi i/k}.$$

A straightforward computation using (25) then gives that the superoperator  $\mathcal{E}_w$  (cf. Eq. (21)) is equal to

$$\mathcal{E}_w(\eta) = \frac{1}{k} \sum_{a \in \mathbb{Z}_k} D_w(a) \eta D_w(a)^\dagger$$

with diagonal Kraus operators

$$D_w(a) |c, d\rangle = \exp \left[ -i\gamma \hat{h}_{u,w}(c - a) - i\gamma \hat{h}_{v,w}(d - a) \right] |c, d\rangle.$$

A general two-qudit mixed state can be classically described by a Hermitian matrix of size  $k^2 \times k^2$ . Multiplying this matrix by a diagonal Kraus operator  $D_w(a)$  takes time  $O(k^4)$ . Thus a single application of the quantum channel  $\mathcal{E}_w$  can be simulated classically in time  $O(k^5)$ . In total, the matrix  $\rho_{u,v}$  can thus be computed in fewer than  $O(k^5(d_u + d_v))$  steps, where  $d_u$  and  $d_v$  are the degrees of  $u$  and  $v$  in the interaction graph defined by  $C$ .

Finally, we claim that the expectation value  $\mu_{u,v}(O)$  in (22) can be computed in time  $O(k^5)$ . Indeed, we have

$$\mu_{u,v}(O) = \text{Tr}(\eta O_{u,v}) \quad \text{where} \quad \eta = B(\beta)^{\otimes 2} \rho_{u,v} B(-\beta)^{\otimes 2}.$$

Let us write

$$\rho_{u,v} = \sum_{a,b \in \mathbb{Z}_k} M(a,b) \otimes |a\rangle\langle b|$$

for some  $k \times k$  matrices  $M(a,b)$ . Accordingly,

$$(B(\beta) \otimes I) \rho_{u,v} (B(-\beta) \otimes I) = \sum_{a,b \in \mathbb{Z}_k} M'(a,b) \otimes |a\rangle\langle b| \equiv \eta'$$

where  $M'(a,b) = B(\beta)M(a,b)B(-\beta)$  can be computed in time  $O(k^3)$  for each pair  $a,b$  by multiplying  $k \times k$  matrices. Thus one can compute the matrix of  $\eta'$  in the  $Z$ -basis in time  $O(k^5)$ . Rewrite this matrix as

$$\eta' = \sum_{a,b \in \mathbb{Z}_k} |a\rangle\langle b| \otimes L(a,b)$$

for some  $k \times k$  matrices  $L(a,b)$ . Then

$$\eta = (I \otimes B(\beta)) \eta' (I \otimes B(-\beta)) = \sum_{a,b \in \mathbb{Z}_k} |a\rangle\langle b| \otimes L'(a,b),$$

where  $L'(a,b) = B(\beta)L(a,b)B(-\beta)$  can be computed in time  $O(k^3)$  for each pair  $a,b$  by multiplying  $k \times k$  matrices. Thus one can compute the matrix of  $\eta$  in time  $O(k^5)$ , as claimed. Finally  $\mu_{u,v}(O) = \text{Tr}(\eta O_{u,v})$  can be computed in time  $O(k^5)$ .

## 5 Comparison of QAOA, RQAOA and classical algorithms

With the algorithm proposed in Section 4, we have simulated level-1 QAOA and level-1-RQAOA on large problem instances of MAX- $k$ -CUT. Our focus is on (approximate) 3-colorability, i.e.,  $k = 3$ . To compare these hybrid algorithms to classical algorithms, we have also applied the best known efficient classical approximation algorithm (as measured by the approximation ratio) to each problem instance, see Section 1 for an overview of these algorithms.

Concretely, we generate random  $d$ -regular, 3-colorable connected graphs on  $n$  vertices according to an ensemble  $\mathcal{G}[d, n]$  defined as follows. Here we assume  $d < 2n/3$  to be even, and  $n$  to be a multiple of 3. A graph  $G$  is drawn from  $\mathcal{G}[d, n]$  as follows:

1. Define a partition  $[n] = V_1 \cup V_2 \cup V_3$  into three pairwise disjoint subsets of each size  $|V_j| = n/3$  for  $j = 1, 2, 3$ .
2. For each pair  $r < s$  with  $r, s \in \{1, 2, 3\}$ , generate a random bipartite  $d$ -regular graph with vertex set  $V_r \cup V_s$  and bipartition  $V_r : V_s$ . This graph is generated by iteratively going through each  $v \in V_r$ , and adding edges  $(v, w_1), \dots, (v, w_d)$  where  $\{w_1, \dots, w_d\} \subset V_s$  are chosen uniformly at random among those vertices in  $V_s$  that (currently) have degree less than  $d$ . If at some point during this process, no such vertices are available, the generation of the bipartite graph is restarted.
3. Check whether the obtained graph contains a complete graph on 3 vertices (i.e., a triangle) and is connected. If either of these properties is not satisfied, start over.

By definition, a graph  $G = (V, E)$  drawn from  $\mathcal{G}[d, n]$  is 3-colorable and thus the cutsizes of the maximum 3-cut is equal to  $C_{\max} = |E| = nd/2$ . In particular, expected approximation ratios for QAOA<sub>1</sub> can be immediately computed from the expectation  $\langle \psi | C | \psi \rangle$  of the cost function Hamiltonian. Similarly, for any approximate coloring  $x \in \mathbb{Z}_3^n$  produced by RQAOA<sub>1</sub> (or any algorithm, for that matter), the achieved approximation ratio is  $C(x)/C_{\max}$ . We use the efficient classical algorithm by Newman [17] for comparison, as it has the best approximation guarantee (worst-case bound) among all known efficient classical algorithms for  $k = 3$ , see Section 1. Note that the theoretical lower bound 0.836008 on the expected approximation ratio of the classical algorithm [17] used here matches the one established for the algorithm in [14] in the case  $k = 3$  and the algorithm in [9].

**Details for implementation.** We empirically observed that the energy landscape contains local maxima and stationary points which often prevent gradient descent in QAOA and RQAOA from finding the optimal solution. The presence of many points with a negligible gradient is a well-studied phenomenon under the name of barren plateaus [15]. Several promising strategies for alleviating this problem have been previously explored [18, 24, 19].

For the specific case of level-1 RQAOA applied to MAX-3-CUT, the energy function to be optimized depends on four arguments, three of which can be efficiently

computed if the fourth one is known. This makes grid search an attractive and cheap alternative to gradient descent as there is only one dimension to be searched. It allows us to sidestep potential convergence problems with gradient descent and to study large graphs. In more detail, the energy  $E(\beta, \gamma) = \langle \psi(\beta, \gamma) | C | \psi(\beta, \gamma) \rangle$  of the cost function Hamiltonian is a function of  $\beta = (\beta_0, \beta_1, \beta_2) \in \mathbb{R}^3$  and  $\gamma \in \mathbb{R}$ . However, as we show in Appendix A, for any fixed value  $\gamma \in \mathbb{R}$ , parameters  $\beta \in \mathbb{R}^3$  maximizing the function  $\beta \mapsto E(\beta, \gamma)$  can be found efficiently by computing roots of a degree-4 polynomial and performing a binary search on the unit circle. This significantly reduces the dimensionality of the optimization problem: only an interval of values  $\gamma \in \mathbb{R}$  needs to be searched. Because the function to be optimized further satisfies  $E((\beta_0, \beta_1, \beta_2), \gamma) = E((- \beta_0, - \beta_2, - \beta_1), - \gamma)$  as  $C$  is self-adjoint and real, and  $\overline{B((\beta_0, \beta_1, \beta_2))} = B((- \beta_0, - \beta_1, - \beta_2))$ , it further suffices to restrict the grid search to the interval  $\gamma \in [0, \pi)$ .

We thus chose 50 equidistant grid points  $\gamma_1, \dots, \gamma_{50}$  in the interval  $[0, \pi]$  for  $\gamma$ . After finding the grid point  $\gamma_s$  that minimizes the cost function (see Appendix A), we performed another refined grid search with 50 additional equidistant grid points on the interval  $[\gamma_{s-1}, \gamma_{s+1}]$ .

In our numerical experiments we find that optimal angles for QAOA<sub>1</sub> for the considered random ensembles of graphs concentrate around certain values. This is in line with the analytical findings of [2] for QAOA: there it is shown that for random ensembles of 3-regular graphs, frequencies of certain local subgraphs concentrate, yielding a simple dependence of the figure of merit on the angles. Similar observations are also used in the proof of our Theorem 3.1.

**Choice of parameters.** For each pair of parameters  $(n, d) \in \{30, 60, 150, 300\} \times \{4, 6, 8, 10\}$ , we generated 20 graphs from  $\mathcal{G}[d, n]$ .

**Numerical results.** In Figures 2, 3, 4 and 5, we illustrate obtained achieved approximation ratios of QAOA<sub>1</sub>, RQAOA<sub>1</sub> and the best approximation ratio of the classical algorithm by Newman [17] over 100 samples generated in the correlation rounding step (magenta, green and blue bars). For Newman’s algorithm, we additionally provide the empirical average and standard deviation of the samples, which are indicated through error bars. Also, the theoretically expected approximation ratio of  $\alpha_{\text{Newman}} = 0.836008$  for Newman’s algorithm is shown.

As expected, we find that RQAOA<sub>1</sub> significantly outperforms QAOA<sub>1</sub> for the considered family of graphs. Their performance deteriorates with increasing degree  $d$  (although RQAOA<sub>1</sub> is less susceptible to this). This is consistent with our no-go result (Theorem 3.1), which applies to random  $d$ -regular graphs of sufficiently large (but also constant) degree  $d$ . We note, however, that Theorem 3.1 does not immediately pertain to the numerical examples considered here because of the (non-explicit) lower bound on  $d$  and its asymptotic nature (as a function of  $n$ ).

More interesting is the comparison to Newman’s classical algorithm. Being a randomized algorithm, it is natural to consider the empirical means and variance of the achieved approximation ratio (although in practice, only the best coloring among a constant number of runs would be used). An analytic understanding of the exact behavior of Newman’s algorithm appears to be a difficult problem. As

already pointed out by Goemans and Williamson in their seminal paper [10], even computing the variance analytically appears to be challenging.

In our numerical experiments we find that the variance of Newman’s algorithm increases with  $d$  and decreases with  $n$ . In cases where the variance is large, among the 100 trials considered, we typically find an essentially optimal solution<sup>1</sup>. We note, however, that the empirical average approximation ratio appears to be close to the theoretical worst-case guarantee  $\alpha_{\text{Newman}}$ .

With the observations above we can distinguish two regimes when comparing the performance of RQAOA to Newman’s algorithm:

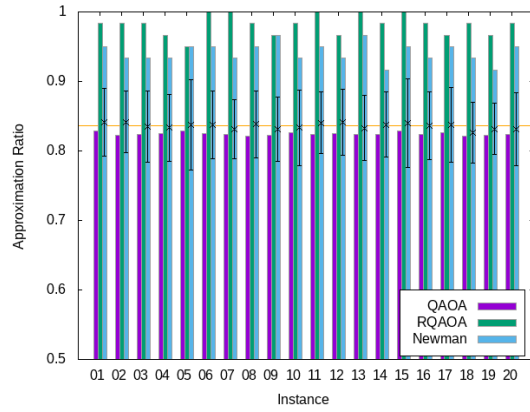
In the first regime (see e.g., Figs. 3c and 4d), where Newman’s algorithm has high variance, we find that the optimal solution returned is superior to the one provided by RQAOA. Nevertheless and perhaps surprisingly, the coloring output by RQAOA (which is a deterministic algorithm up to estimation errors for expectation values when used in practice) outperforms the average approximation ratio of Newman’s algorithm.

In the second regime (see e.g., Figs. 2a and 5a), Newman’s algorithm is strongly concentrated about its average. In this case, RQAOA<sub>1</sub> outperforms virtually all instances returned by Newman’s algorithm. This regime includes instances with a large number of vertices, indicating that RQAOA may be a useful heuristic algorithm for problems of practical interest.

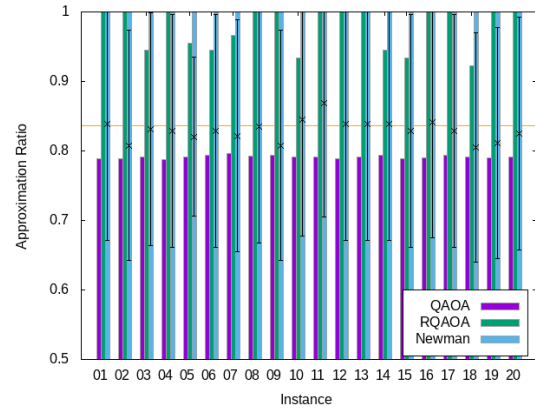
Larger levels  $p > 1$  may further increase achieved approximation ratios of RQAOA. Unfortunately, exploring this may require new ideas or actual quantum devices. This is suggestive of RQAOA being a promising candidate for NISQ applications.

---

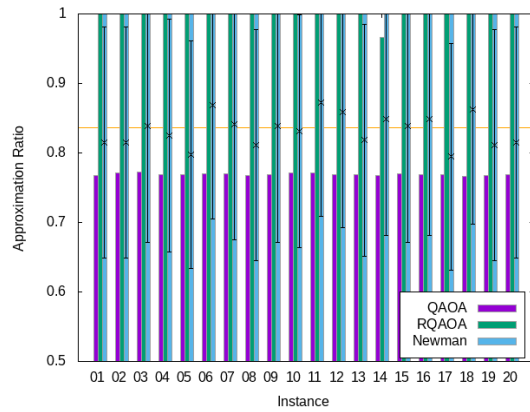
<sup>1</sup>Let us mention that the 3-coloring problem for dense 3-colorable graphs admits an efficient algorithm [1], although this is not directly connected to the performance of Newman’s algorithm.



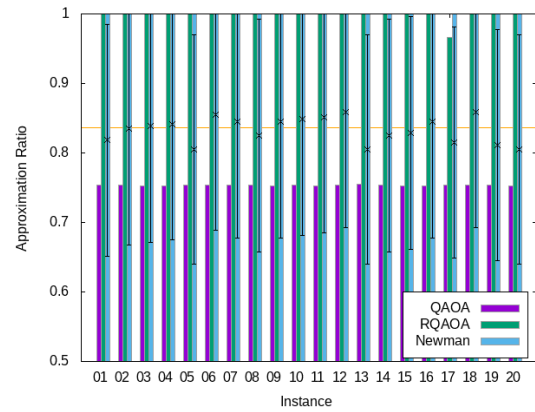
(a)  $n = 30, d = 4$



(b)  $n = 30, d = 6$

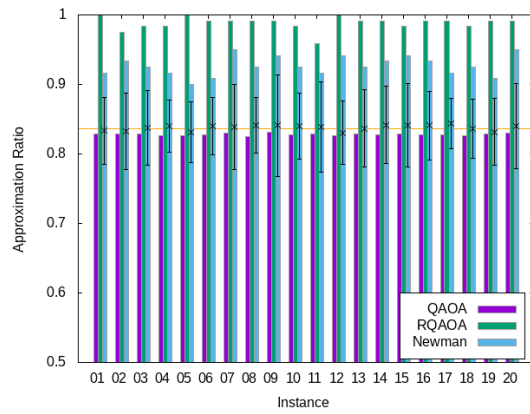


(c)  $n = 30, d = 8$

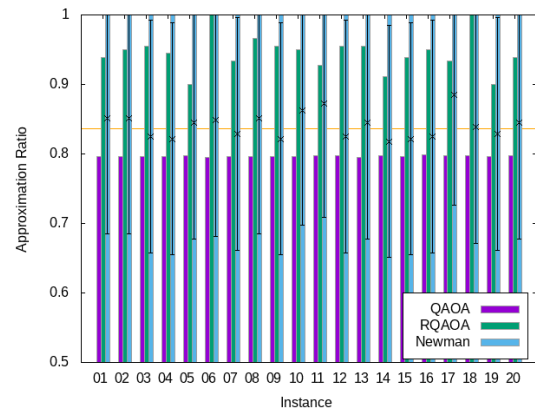


(d)  $n = 30, d = 10$

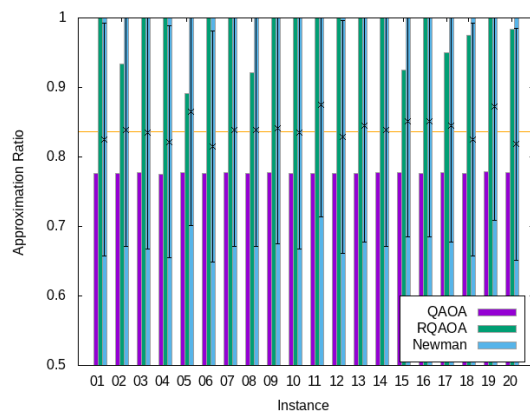
Figure 2: Comparison of approximation ratios between  $\text{QAOA}_1$ ,  $\text{RQAOA}_1$  and the algorithm by Newman for MAX-3-CUT, where we took the best approximation ratio over 100 samples of graphs with  $n = 30$  vertices. The expected approximation ratio of Newman's algorithm is indicated by the horizontal line at  $\alpha = 0.836008$ . For each graph, the empirical mean and standard deviation of Newman's algorithm are indicated through the error bars.



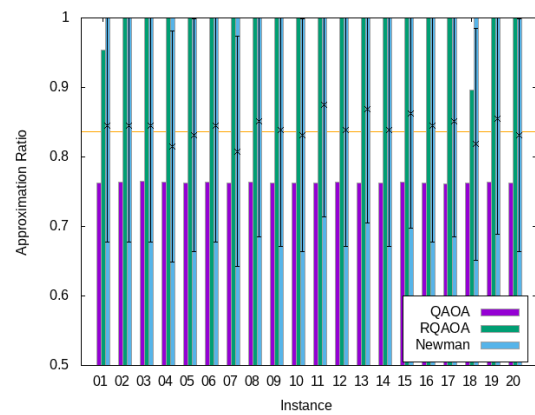
(a)  $n = 60, d = 4$



(b)  $n = 60, d = 6$



(c)  $n = 60, d = 8$



(d)  $n = 60, d = 10$

Figure 3: Comparison of approximation ratios between  $\text{QAOA}_1$ ,  $\text{RQAOA}_1$  and the algorithm by Newman for MAX-3-CUT, where we took the best approximation ratio over 100 samples of graphs with  $n = 60$  vertices. The expected approximation ratio of Newman's algorithm is indicated by the horizontal line at  $\alpha = 0.836008$ . For each graph, the empirical mean and standard deviation of Newman's algorithm are indicated through the error bars.

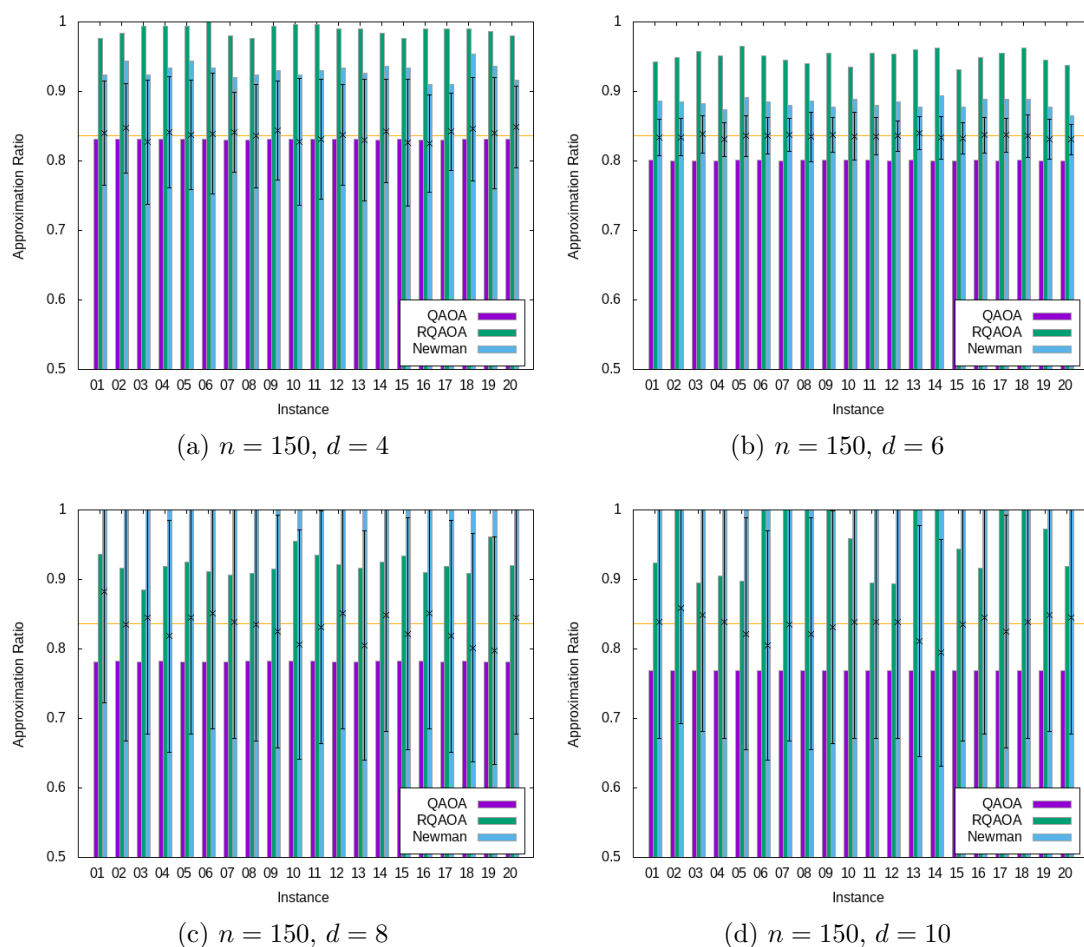
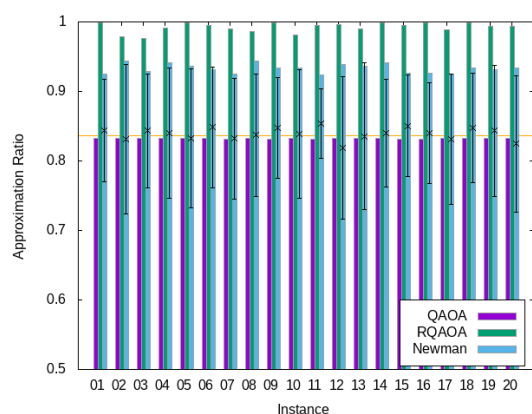
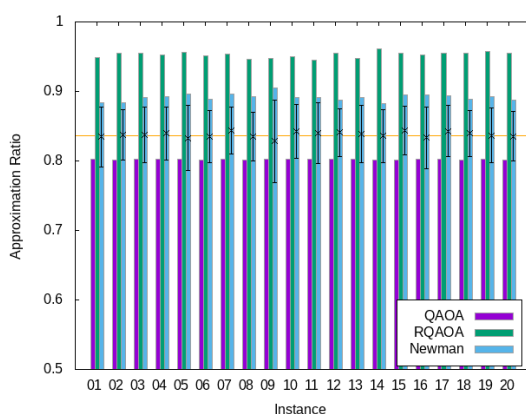


Figure 4: Comparison of approximation ratios between  $\text{QAOA}_1$ ,  $\text{RQAOA}_1$  and the algorithm by Newman for MAX-3-CUT, where we took the best approximation ratio over 100 samples of graphs with  $n = 150$  vertices. The expected approximation ratio of Newman's algorithm is indicated by the horizontal line at  $\alpha = 0.836008$ . For each graph, the empirical mean and standard deviation of Newman's algorithm are indicated through the error bars.

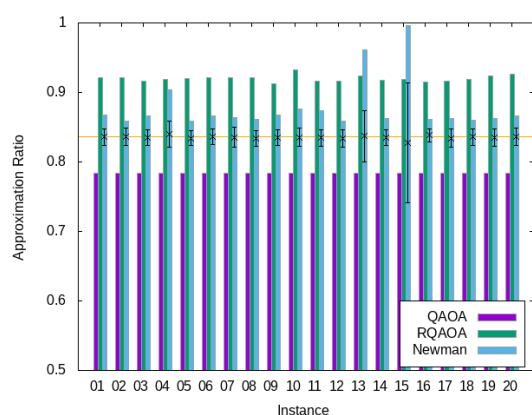




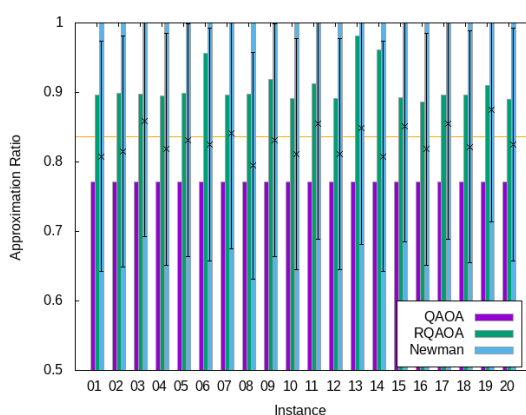
(a)  $n = 300, d = 4$



(b)  $n = 300, d = 6$



(c)  $n = 300, d = 8$



(d)  $n = 300, d = 10$

Figure 5: Comparison of approximation ratios between  $\text{QAOA}_1$ ,  $\text{RQAOA}_1$  and the algorithm by Newman for MAX-3-CUT, where we took the best approximation ratio over 100 samples of graphs with  $n = 300$  vertices. The expected approximation ratio of Newman's algorithm is indicated by the horizontal line at  $\alpha = 0.836008$ . For each graph, the empirical mean and standard deviation of Newman's algorithm are indicated through the error bars.

## 6 Discussion and Outlook

We formulated a variation of QAOA using qudits which is applicable to non-binary combinatorial optimization. We study its applicability by considering the MAX- $k$ -CUT problem with  $k > 2$ . We show that this version of QAOA shares certain limitations with the original proposal for qubits: its expected performance on random  $d$ -regular  $k$ -colorable graphs (for  $d = o(\sqrt{n})$ ) approaches random guessing in the limit of large  $d$ . We give an efficient classical simulation algorithm for its level-1 version. This method extends to problem Hamiltonians with pairwise commuting 2-local terms. The algorithm can also be used to simulate level-1 recursive QAOA (RQAOA), a hybrid classical-quantum algorithm designed to overcome the limitations of standard QAOA.

While efficient simulability precludes the possibility of a quantum advantage compared to the best efficient classical algorithm, it offers a crucial window into the performance of QAOA and RQAOA. We use our simulation algorithm to perform numerical experiments on graphs with up to 300 vertices, obtaining approximation ratios for MAX-3-CUT achieved by level-1 QAOA and RQAOA. We compare these results to the best known efficient classical algorithms for MAX-3-CUT, specifically the recent algorithm by Newman [17]. Our experiments show that RQAOA<sub>1</sub> (which significantly outperforms QAOA<sub>1</sub>) is competitive with Newman’s classical algorithm for generic  $k$ -colorable graphs. See Section 5 for a detailed discussion.

Our observations suggest that RQAOA may be a viable application for NISQ devices combined with efficient classical computation: it may be applicable to problem sizes of real-world relevance. Our results are indicative of the potential of RQAOA at larger levels  $p$ , which may no longer be simulable classically, and which hold promise. Future work may seek to establish performance guarantees for RQAOA <sub>$p$</sub>  akin to the rigorous results available for existing classical algorithms. As an example, we proved in [3] that for the “ring of disagrees”, RQAOA<sub>1</sub> achieves the optimal approximation ratio. A next step in the analysis of RQAOA would be to find more interesting classes of graphs for which achievability results can be established.

**Acknowledgments.** This work is supported in part by the Army Research Office under Grant Number W911NF-20-1-0014. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein. RK and AK gratefully acknowledge support by the DFG cluster of excellence 2111 (Munich Center for Quantum Science and Technology). ET acknowledges funding provided by DOE Award Number: DE-SC0018407 Quantum Error Correction and Spacetime Geometry and the Institute for Quantum Information and Matter, an NSF Physics Frontiers Center (NSF Grant PHY-1733907).

## A MAX-3-CUT angle optimization without gradient descent

First let us write the expectation value (22) in the form that makes its dependence on  $\beta$  more explicit. Recall from (6) that for each  $a \in \mathbb{Z}_k$ , the  $X$ -eigenstate  $|\phi_a\rangle \equiv Z^a|+\rangle$  is an eigenvector of the unitary  $B(\beta)$  to eigenvalue  $e^{i\beta a}$ . Since the vectors  $\{|\phi_a\rangle\}_{a \in \mathbb{Z}_k}$  form an orthonormal basis of  $\mathbb{C}^k$ , one gets

$$\mu_{u,v}(O) = \sum_{p,q \in \mathbb{Z}_k} e^{i(\beta_p + \beta_q)} \langle \phi_p \otimes \phi_q | \rho_{u,v} B(-\beta)^{\otimes 2} O_{u,v} | \phi_p \otimes \phi_q \rangle .$$

Consider the special case  $O = Z^r \otimes Z^{-r}$ . The identity  $Z^r |\phi_p\rangle = |\phi_{p+r}\rangle$  gives

$$\mu_{u,v}(Z^r \otimes Z^{-r}) = \sum_{p,q \in \mathbb{Z}_k} e^{i(\beta_p + \beta_q - \beta_{p+r} - \beta_{q-r})} \langle \phi_p \otimes \phi_q | \rho_{u,v} | \phi_{p+r} \otimes \phi_{q-r} \rangle \quad (26)$$

For  $(\beta, \gamma) \in \mathbb{R}^3 \times \mathbb{R}$ , let  $E(\beta, \gamma) = \langle \psi(\beta, \gamma) | C | \psi(\beta, \gamma) \rangle$  denote the energy in the level-1 QAOA state, where the cost function Hamiltonian  $C$  is given by Eq. (24) with  $k = 3$ . Let  $\gamma \in \mathbb{R}$  be fixed in the following. Let us write  $E(\beta) = E(\beta, \gamma)$ . Here we consider the problem of finding  $\max_{\beta} E(\beta)$ .

With Eq. (26) we have

$$E(\beta) = \sum_{1 \leq p < q \leq n} \sum_{a,b,c \in \mathbb{Z}_3} h_{p,q}(c) e^{i(\beta_a + \beta_b - \beta_{a+c} - \beta_{b-c})} \langle \phi_a \otimes \phi_b | \rho_{p,q} | \phi_{a+c} \otimes \phi_{b-c} \rangle .$$

Clearly, the terms with  $c = 0$  do not depend on  $\beta$  and the sum over  $p, q$  gives  $\text{Tr}(\rho_{p,q}) = 1$ . For each  $p < q$  the sum of all terms with  $c = -1$  and the sum of all terms with  $c = 1$  are complex conjugates of each other. Indeed, these sums are expected values of  $h_{p,q}(1) Z_p Z_q^{-1}$  and  $h_{p,q}(1)^* Z_p^{-1} Z_q$ . Here we noted that  $h_{p,q}(-1) = h_{p,q}(1)^*$ . Thus

$$E(\beta) = \sum_{\substack{1 \leq p \\ < q \leq n}} h_{p,q}(0) + \sum_{\substack{a,b \\ \in \mathbb{Z}_3}} 2 \text{Re} \left( h_{p,q}(1) e^{i(\beta_a + \beta_b - \beta_{a+1} - \beta_{b-1})} \langle \phi_a \otimes \phi_b | \rho_{p,q} | \phi_{a+1} \otimes \phi_{b-1} \rangle \right)$$

Let  $\bar{\beta} \equiv \beta_0 + \beta_1 + \beta_2$ . Some simple algebra gives

$$E(\beta) = C + \text{Re} \sum_{a \in \mathbb{Z}_3} g_a e^{i(3\beta_a - \bar{\beta})}$$

where

$$C = \sum_{1 \leq p < q \leq n} h_{p,q}(0) + 2 \text{Re} \sum_{a \in \mathbb{Z}_3} h_{p,q}(1) \langle \phi_a \otimes \phi_{a+1} | \rho_{p,q} | \phi_{a+1} \otimes \phi_a \rangle ,$$

and

$$g_a = 2 \sum_{\substack{1 \leq p \\ < q \leq n}} h_{p,q}(1) \langle \phi_a \otimes \phi_a | \rho_{p,q} | \phi_{a+1} \otimes \phi_{a-1} \rangle + h_{p,q}(1)^* \langle \phi_a \otimes \phi_a | \rho_{p,q} | \phi_{a-1} \otimes \phi_{a+1} \rangle ,$$

Define  $\theta_a = 3\beta_a - \bar{\beta}$ . Then  $\theta_0 + \theta_1 + \theta_2 = 0$ . Thus it suffices to maximize a function

$$F'(\theta) = \operatorname{Re} \sum_{a \in \mathbb{Z}_3} g_a e^{i\theta_a}$$

over  $\theta \in \mathbb{R}^3$  subject to a constraint

$$\theta_0 + \theta_1 + \theta_2 = 0.$$

For fixed  $\theta_0$  the maximum over  $\theta_1$  can be computed analytically using the identity

$$\max_{\theta_1} \operatorname{Re} \left( g_1 e^{i\theta_1} + g_2 e^{-i\theta_0 - i\theta_1} \right) = \max_{\theta_1} \operatorname{Re} \left( e^{i\theta_1} \left( g_1 + g_2^* e^{i\theta_0} \right) \right) = \left| g_1 + g_2^* e^{i\theta_0} \right|.$$

The maximum is achieved at  $\theta_1 = -\arg \left( g_1 + g_2^* e^{i\theta_0} \right)$ . Let  $z \equiv e^{i\theta_0}$ . It remains to maximize a function

$$F''(z) = \max_{\theta \in \mathbb{R}^3} \begin{array}{l} F'(\theta) = \operatorname{Re}(g_0 z) + |g_1 + g_2^* z| \\ \theta_0 + \theta_1 + \theta_2 = 0 \\ e^{i\theta_0} = z \end{array}$$

over the unit circle  $\{z \in \mathbb{C} : |z| = 1\}$ . Fix some  $f \in \mathbb{R}$ . One can easily check that conditions  $F''(z) = f, |z| = 1$  imply  $p_f(z) = 0, |z| = 1$ , where  $p_f(z)$  is a degree-4 polynomial

$$p_f(z) = \left( -\frac{g_0^2}{4} \right) z^4 + (g_1^* g_2^* + f g_0) z^3 + z^2 \left( |g_1|^2 + |g_2|^2 - f^2 - (1/2) |g_0|^2 \right) + z (g_1 g_2 + f g_0^*) - \frac{(g_0^*)^2}{4}$$

For a given  $f$  one can analytically compute roots of  $p_f(z)$ , select roots lying on the unit circle, and check whether at least one of those roots  $z$  satisfies  $F''(z) = f$ . Thus one can check whether an equation  $F''(z) = f$  has solutions  $z$  on the unit circle. Now one can maximize  $F''(z)$  over the unit circle using the binary search over  $f$ . Given the optimal value  $\theta_0 = \arg \max_{\theta} F''(e^{i\theta})$ , one computes  $\theta_1 = -\arg \left( g_1 + g_2^* e^{i\theta_0} \right), \theta_2 = -\theta_0 - \theta_1$ , and solves a linear system  $\theta_a = 3\beta_a - \bar{\beta}$  to find  $\beta_0, \beta_1, \beta_2$ .

## References

- [1] Noga Alon and Nabil Kahale. A spectral technique for coloring random 3-colorable graphs. *SIAM J. Comput.*, 26(6):1733–1748, December 1997. <https://doi.org/10.1137/S0097539794270248>.
- [2] Fernando G. S. L. Brandao, Michael Broughton, Edward Farhi, Sam Gutmann, and Hartmut Neven. For Fixed Control Parameters the Quantum Approximate Optimization Algorithm’s Objective Function Value Concentrates for Typical Instances, 2018. <https://doi.org/10.48550/arXiv.1812.04170>.
- [3] Sergey Bravyi, Alexander Kliesch, Robert Koenig, and Eugene Tang. Obstacles to variational quantum optimization from symmetry protection. *Phys. Rev. Lett.*, 125:260505, Dec 2020. <https://doi.org/10.1103/PhysRevLett.125.260505>.

- [4] Amin Coja-Oghlan, Cristopher Moore, and Vishal Sanwalani. Max- $k$ -cut and approximating the chromatic number of random graphs. *Random Structures & Algorithms*, 28(3):289–322, 2006. <https://doi.org/10.1002/rsa.20096>.
- [5] Daniel J. Egger, Jakub Mareček, and Stefan Woerner. Warm-starting quantum optimization. *Quantum*, 5:479, June 2021. <https://doi.org/10.22331/q-2021-06-17-479>.
- [6] Edward Farhi, David Gamarnik, and Sam Gutmann. The Quantum Approximate Optimization Algorithm Needs to See the Whole Graph: Worst Case Examples, 2020. <https://doi.org/10.48550/arXiv.2005.08747>.
- [7] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A Quantum Approximate Optimization Algorithm, 2014. <https://doi.org/10.48550/arXiv.1411.4028>.
- [8] Alan Frieze and M. Jerrum. Improved Approximation Algorithms for MAX  $k$ -CUT and MAX BISECTION. *Algorithmica*, 18(1):67–81, 05 1997. <https://doi.org/10.1007/BF02523688>.
- [9] Michel X. Goemans and David Williamson. Approximation Algorithms for MAX-3-CUT and Other Problems via Complex Semidefinite Programming. In *Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing*, STOC '01, page 443–452, New York, NY, USA, 2001. Association for Computing Machinery. <https://doi.org/10.1145/380752.380838>.
- [10] Michel X. Goemans and David P. Williamson. Improved Approximation Algorithms for Maximum Cut and Satisfiability Problems Using Semidefinite Programming. *J. ACM*, 42(6):1115–1145, 1995. <https://doi.org/10.1145/227683.227684>.
- [11] Jun-Dong Cho, S. Raje, and M. Sarrafzadeh. Fast Approximation Algorithms on Maxcut,  $k$ -Coloring, and  $k$ -Color Ordering for VLSI Applications. *IEEE Transactions on Computers*, 47(11):1253–1266, 1998. <https://doi.org/10.1109/12.736440>.
- [12] Viggo Kann, Sanjeev Khanna, Jens Lagergren, and Alessandro Panconesi. On the Hardness of Approximating Max  $k$ -Cut and its Dual. *Chicago Journal of Theoretical Computer Science*, 1997:1–18, June 1997. <https://doi.org/10.4086/cjtc.1997.002>.
- [13] Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan O’Donnell. Optimal Inapproximability Results for MAX-CUT and Other 2-Variable CSPs? *SIAM Journal on Computing*, 37(1):319–357, 2007. <https://doi.org/10.1137/S0097539705447372>.
- [14] Etienne Klerk, Dmitrii Pasechnik, and J.P. Warners. On Approximate Graph Colouring and MAX- $k$ -CUT Algorithms Based on the  $\vartheta$ -Function. *Journal of Combinatorial Optimization*, 8:267–294, 09 2004. <https://doi.org/10.1023/B:JOCO.0000038911.67280.3f>.
- [15] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nature Communications*, 9(4812):1–6, 2018. <https://doi.org/10.1038/s41467-018-07090-4>.

- [16] Jarrod R. McClean, Matthew P. Harrigan, Masoud Mohseni, Nicholas C. Rubin, Zhang Jiang, Sergio Boixo, Vadim N. Smelyanskiy, Ryan Babbush, and Hartmut Neven. Low-depth mechanisms for quantum optimization. *PRX Quantum*, 2:030312, Jul 2021. <https://doi.org/10.1103/PRXQuantum.2.030312>.
- [17] Alantha Newman. Complex Semidefinite Programming and Max-k-Cut. In *1st Symposium on Simplicity in Algorithms (SOSA 2018)*, volume 61 of *OpenAccess Series in Informatics (OASICs)*, pages 13:1–13:11, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. <https://doi.org/10.4230/OASICs.SOSA.2018.13>.
- [18] J.S. Otterbach, R. Manenti, N. Alidoust, A. Bestwick, M. Block, B. Bloom, S. Caldwell, N. Didier, E. Schuyler Fried, S. Hong, et al. Un-supervised Machine Learning on a Hybrid Quantum Computer, 2017. <https://doi.org/10.48550/arXiv.1712.05771>.
- [19] Ruslan Shaydulin, Ilya Safro, and Jeffrey Larson. Multistart Methods for Quantum Approximate Optimization. In *2019 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1–8. IEEE, 2019. <https://doi.org/10.1109/HPEC.2019.8916288>.
- [20] Reuben Tate, Majid Farhadi, Creston Herold, Greg Mohler, and Swati Gupta. Bridging Classical and Quantum with SDP initialized warm-starts for QAOA, 2020. <https://doi.org/10.48550/arXiv.2010.14021>.
- [21] Paul M. B. Vitányi. How well can a graph be  $n$ -colored? *Discrete Math.*, 34(1):69–80, Jan 1981. [https://doi.org/10.1016/0012-365X\(81\)90023-6](https://doi.org/10.1016/0012-365X(81)90023-6).
- [22] Zhihui Wang, Stuart Hadfield, Zhang Jiang, and Eleanor G. Rieffel. Quantum approximate optimization algorithm for MaxCut: A fermionic view. *Phys. Rev. A*, 97:022304, Feb 2018. <https://doi.org/10.1103/PhysRevA.97.022304>.
- [23] Nicholas C. Wormald. The asymptotic distribution of short cycles in random regular graphs. *Journal of Combinatorial Theory, Series B*, 31(2):168 – 182, 1981. [https://doi.org/10.1016/S0095-8956\(81\)80022-6](https://doi.org/10.1016/S0095-8956(81)80022-6).
- [24] Jiahao Yao, Marin Bukov, and Lin Lin. Policy Gradient based Quantum Approximate Optimization Algorithm, 2020. <https://doi.org/10.48550/arXiv.2002.01068>.

## A.2 Twisted hybrid algorithms for combinatorial optimization

# Twisted hybrid algorithms for combinatorial optimization

Libor Caha, Alexander Kliesch, Robert Koenig

---

The two currently best known classical approximation algorithms for the MAX-CUT problem restricted to 3-regular graphs by Feige, Karpinski, and Langberg (FKL) [15] and Halperin, Livnat, and Zwick (HLZ) [16] both utilize post-processing: they first produce an intermediate cut (in their case, by relaxing the combinatorial optimization problem to a semidefinite program) and then improve upon it via classical, efficient methods. In Core Article II), we investigate the effect of their respective post-processing methods on QAOA applied to the MAX-CUT problem and use them to modify (“twist”) the algorithm. We note that our techniques are, in principle, applicable to any variational quantum algorithm.

## A.2.1 Main Results

A naive application of post-processing would consist of running the standard QAOA and then applying the corresponding techniques on the resulting cut. However, both the FKL and HLZ post-processing techniques allow for a rigorous, analytical lower bound on the approximation ratio achieved by the total algorithm. The key idea is to realize that these improvements can be modeled in terms of additional terms in the cost Hamiltonian used in the description of QAOA. We therefore optimize the angles in QAOA with respect to this modified Hamiltonian instead of the usual problem Hamiltonian to account for the post-processing that we apply to the cut resulting from preparing and measuring the QAOA state. In particular, the new Hamiltonian is still a local Hamiltonian for both post-processing methods, allowing us to use established techniques exploiting the locality of QAOA to derive provable, analytical lower bounds on the approximation ratio of the twisted version of QAOA. We observe that, despite not using any more quantum resources, twisted QAOA of level  $p$  improves upon standard QAOA of level  $p$  and is in fact competitive with the performance of standard QAOA of level  $p + 1$ .

## A.2.2 Individual Contribution

I am the principal author of this article. The idea for this work came after discussions with my doctoral advisor Robert Koenig and Libor Caha, who was then visiting our department M5 before accepting a position at the chair a few months later; we then started to investigate the potential of the idea more thoroughly. I was significantly involved in finding the ideas of the article, in particular in establishing the lower bounds, and was also significantly involved in writing the article with the exception of Section 3.



# Permission to include:

Libor Caha, Alexander Kliesch, Robert Koenig  
“Twisted hybrid algorithms for combinatorial optimization”  
*Quantum Science and Technology* 7, 045013 (2022)

**IOP Publishing**

Dear Mr Alexander Kliesch,

Re: Twisted hybrid algorithms for combinatorial optimization

Article reference: *QST-101714*

IOP Publishing and your institution/funder have an open access agreement that promotes sustainable open access and may cover the article publication charge (APC) for your paper in *Quantum Science and Technology*.

We have therefore made your article open access to take advantage of this agreement. It will be published with a CC-BY licence. Please choose the Gold Open Access option when completing your Assignment of Copyright form.

← → ↻ 🏠 🔒 <https://publishingsupport.iopscience.iop.org/reusing-iop-published-content/> 📄 150% ★ ☰

## When permission is not needed:

There are a few instances in which you don't need permission to reuse IOP published content:

- If you are publishing a **new work with IOP** and wish to reuse content from an **IOP journal or eBook, or a partner journal that we handle permissions for**. IOP already has the permission to use this content, so you do not need to request it again.
- If the content was published on an **open-access** basis that allows the type of reuse you need. For details of open access licences, please see [Creative Commons Licences](#).
- If you are the **original author** of the content you wish to reuse, and your reuse falls within the allowances of our [Author Rights policy](#).

← → ↻ 🏠 🔒 <https://publishingsupport.iopscience.iop.org/creative-commons-licences/> 📄 130% ☆ ☰

## Can I reuse without permission?

- **CC BY** – This licence lets others distribute, remix, tweak and build upon your work, even commercially, as long as they credit you for the original creation
  - Yes, you can reuse without permission

PAPER • OPEN ACCESS

## Twisted hybrid algorithms for combinatorial optimization

To cite this article: Libor Caha *et al* 2022 *Quantum Sci. Technol.* **7** 045013

View the [article online](#) for updates and enhancements.

### You may also like

- [Computational phase transitions: benchmarking Ising machines and quantum optimisers](#)  
Hariphan Philathong, Vishwa Akshay, Ksenia Samburskaya *et al.*
- [Hybrid quantum–classical optimization with cardinality constraints and applications to finance](#)  
Samuel Fernández-Lorenzo, Diego Porras and Juan José García-Ripoll
- [Domain wall encoding of discrete variables for quantum annealing and QAOA](#)  
Nicholas Chancellor



**IOP | ebooks™**

Bringing together innovative digital publishing with leading authors from the global scientific community.

Start exploring the collection—download the first chapter of every title for free.

# Quantum Science and Technology



PAPER

## Twisted hybrid algorithms for combinatorial optimization

OPEN ACCESS

RECEIVED  
22 March 2022REVISED  
24 June 2022ACCEPTED FOR PUBLICATION  
6 July 2022PUBLISHED  
12 August 2022Libor Caha<sup>1,2</sup> , Alexander Kliesch<sup>1,2,\*</sup>  and Robert Koenig<sup>1,2</sup><sup>1</sup> Zentrum Mathematik, Technical University of Munich, Germany<sup>2</sup> Munich Center for Quantum Science and Technology, Munich, Germany

\* Author to whom any correspondence should be addressed.

E-mail: [libor.caha@tum.de](mailto:libor.caha@tum.de), [kliesch@ma.tum.de](mailto:kliesch@ma.tum.de) and [robert.koenig@tum.de](mailto:robert.koenig@tum.de)**Keywords:** quantum algorithms, variational quantum algorithms, combinatorial optimization, approximation algorithms, hybrid algorithms, MaxCut

Original content from this work may be used under the terms of the [Creative Commons Attribution 4.0 licence](https://creativecommons.org/licenses/by/4.0/).

Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.



### Abstract

Proposed hybrid algorithms encode a combinatorial cost function into a problem Hamiltonian and optimize its energy by varying over a set of states with low circuit complexity. Classical processing is typically only used for the choice of variational parameters following gradient descent. As a consequence, these approaches are limited by the descriptive power of the associated states. We argue that for certain combinatorial optimization problems, such algorithms can be hybridized further, thus harnessing the power of efficient non-local classical processing. Specifically, we consider combining a quantum variational ansatz with a greedy classical post-processing procedure for the MaxCut-problem on three-regular graphs. We show that the average cut-size produced by this method can be quantified in terms of the energy of a modified problem Hamiltonian. This motivates the consideration of an improved algorithm which variationally optimizes the energy of the modified Hamiltonian. We call this a twisted hybrid algorithm since the additional classical processing step is combined with a different choice of variational parameters. We exemplify the viability of this method using the quantum approximate optimization algorithm (QAOA), giving analytic lower bounds on the expected approximation ratios achieved by twisted QAOA. We observe that for levels  $p = 1, \dots, 5$ , these lower bounds are comparable to the known lower bounds on QAOA at level  $p + 1$  for high-girth graphs. This suggests that using twisted QAOA can reduce the circuit depth by 4 and the number of variational parameters by 2.

### 1. Introduction

Due to their real-world interest, problems and algorithms for combinatorial optimization figure prominently in present-day theoretical computer science. For theoretical physics, the profound and immediate connections to the physics, e.g., of Ising or Potts models are particularly appealing. Combinatorial optimization also provides an intriguing potential area of application of near-term quantum devices with clear figures of merit such as approximation ratios. Yet the study of quantum algorithms for these problems is still in its infancy, especially when compared to the intensely studied area of classical algorithms. For example, for classical algorithms, an established bound [13, 14] on efficiently achievable approximation ratios for MaxCut under the unique games conjecture matches that achieved by the celebrated Goemans–Williamson algorithm [10] (see also [4]). It appears rather unlikely that under the unique games conjecture an efficient quantum algorithm can outperform the Goemans–Williamson algorithm for generic graphs. Even the more modest goal of identifying special families of instances for which a quantum algorithm outperforms comparable efficient classical algorithms appears to be out of reach. Independently of whether or not one can find a provable real-world quantum advantage in the setting of combinatorial optimization, or ends up using quantum devices as a heuristic to efficiently find approximate solutions, or finds novel classical algorithms inspired by quantum ones (as has happened before), it is natural to study to what extent existing proposals can be improved in a systematic manner with

associated performance guarantees. This is what we pursue here in the context of hybrid classical-quantum algorithms.

For the problem of finding (or approximating) the maximum of a combinatorial cost function  $C : \{0, 1\}^n \rightarrow \mathbb{R}$  (given by polynomially many terms), typical hybrid algorithms proceed by defining the cost function Hamiltonian

$$H_C = \sum_{z \in \{0,1\}^n} C(z) |z\rangle\langle z|$$

in terms of local terms, and a parametrized family  $\{U_G(\theta)\}_{\theta \in \Theta}$  of  $n$ -qubit unitary circuits. The latter might be parametrized by the underlying graph of the cost function or in case of hardware-efficient algorithms tailored to the physical device [12]. The parametrized family gives rise to variational ansatz states

$$|\Psi(\theta)\rangle = U_G(\theta)|0\rangle^{\otimes n},$$

that can be prepared with  $U_G(\theta)$  from a product state  $|0\rangle^{\otimes n}$ . Measuring  $\Psi(\theta)$  in the computational basis then provides a sample  $z \in \{0, 1\}^n$  from the distribution  $p(z) = |\langle z|\Psi(\theta)\rangle|^2$  such that the expectation value of the associated cost function is equal to the energy  $\mathbb{E}[C(z)] = \langle \Psi(\theta) | H_C | \Psi(\theta) \rangle$  of the state  $\Psi(\theta)$  with respect to  $H_C$ . Thus the problem of maximizing  $C$  is translated to that of finding a value of the (vector of) parameters  $\theta$  maximizing the energy of  $\Psi(\theta)$ . The latter step is envisioned to be performed e.g., by numerical gradient descent or a similar classical procedure prescribing (iteratively) what parameters  $\theta$  to try. The computation of this prescription (according to obtained measurement results) is the classical processing part of the quantum algorithm leading to the term *hybrid*. We will refer to this form of algorithm as a ‘bare’ hybrid algorithm in the following.

The potential utility of this approach hinges on a number of factors. Of primary importance—beyond questions of convergence or efficiency—is whether the family  $\{\Psi(\theta)\}_{\theta \in \Theta}$  of states is sufficiently rich to variationally capture the (classical) correlations of high-energy states of  $H_C$ . There is an inherent tension here between the requirement of applicability using near-term devices, and the descriptive power, i.e., required complexity of these states: on the one hand, each unitary  $U_G(\theta)$  is supposed to be realized by a low-depth circuit with local gates (making it amenable to experimental realization on a near-term device), and the dimensionality of the parameter or ‘search’ space  $\Theta$  should be low to guarantee fast convergence e.g., of gradient descent. On the other hand, states having high energy with respect to  $H_C$  and belonging to the considered family of variational states may have intrinsically high circuit complexity, and, correspondingly, may also require a large number of variational parameters to approximate. The unavoidability of this issue has been demonstrated using the MaxCut-problem on expander graphs with  $n$  vertices and the quantum approximate optimization algorithm (QAOA) at level  $p$ : here the parameter space is  $\Theta = [0, 2\pi)^{2p}$  and the corresponding circuits  $U_G(\theta)$  have depth  $O(pd)$ . Locality and symmetry of the ansatz imply that achievable expected approximation ratios are upper bounded by a constant (below that achieved by Goemans–Williamson) unless  $p = \Omega(\log n)$  [3]. In fact, the locality of the ansatz alone implies that for smaller values of  $p$ , the achieved expected approximation ratio is not better than of a random guessing for random bipartite graphs, as shown in [6].

These fundamental limitations of ‘standard’ hybrid algorithms are tied to the assumption that an increased complexity of the required quantum operations is unacceptable and/or infeasible in the near term. Under these circumstances, the only way forward appears to be to use alternative, possibly more powerful (e.g., non-local) efficient classical processing which could exploit the limited available quantum resources more effectively. One example where a classical post-processing is used is [7], where QAOA is combined with a greedy ‘pruning’ method to produce an independent set of large size. Here post-processing is needed, in particular, to ensure that the output is indeed an independent set. Another proposal in this direction is the idea of ‘warm-starting’ QAOA with a solution provided by the Goemans–Williamson algorithm [5] (see also [16]). The warm-starting approach has the appeal that—by construction—the Goemans–Williamson approximation ratio can be guaranteed in this approach (assuming convergence of the energy optimization). An alternative is the recursive QAOA (RQAOA) method [2, 3] which uses QAOA states to iteratively identify variables to eliminate. This effectively reduces the problem size but increases the connectivity and thus the circuit complexity of the iteratively obtained subproblems. Furthermore, analytical bounds on the expected approximation ratios are unknown except for very special examples [3]. For both warm-starting QAOA as well as RQAOA, one deviates from the original QAOA ansatz, leading to different variational states and corresponding quantum circuits.

### 1.1. Our contribution

**Basic idea.** Here we consider arguably more minimal adaptations of hybrid variational algorithms for the MaxCut-problem on three-regular graphs. For a given bare hybrid algorithm  $\mathcal{A}$  involving a family

$\{\Psi(\theta)\}_{\theta \in \Theta}$  of variational ansatz states as described above, we show how to construct a modified algorithm  $\mathcal{A}^+$  which uses the *same* family of states  $\{\Psi(\theta)\}_{\theta \in \Theta}$ . The algorithm  $\mathcal{A}^+$  will be called *twisted- $\mathcal{A}$* . Our modified algorithms are directly motivated by the work of Feige, Karpinski, and Langberg [9] (referred to as FKL in the following). These authors propose an algorithm for the MaxCut problem on three-regular graphs which proceeds by solving a semidefinite program relaxation (similar to Goemans and Williamson), and subsequently improving the rounded solution by a simple greedy post-processing technique. We also consider the improved version by Halperin, Livnat, and Zwick [11] (referred to as HLZ below) which involves a more non-local greedy procedure. For some motivation, see the following example.

**Example.** Consider a simple motivational example of a greedy post-processing procedure that can improve a given cut. The input will be a three-regular graph  $G = (V, E)$  and a cut  $C$ . We say that a vertex is unsatisfied when all three of its neighbours lie in the same partition of the cut as it does. The algorithm will repeatedly run through the vertices and check whether some of them are unsatisfied. If it finds an unsatisfied vertex it moves it to the opposite side of the cut and repeats the process with the updated cut until none of the vertices is unsatisfied. Since moving one vertex increases the cut size by 3 and potentially lowers the number of unsatisfied vertices by 4, one can show that this procedure improves the cut size by at least  $\frac{3}{4}$  times the number of unsatisfied vertices in the initial cut. Let us apply this greedy procedure to a random cut, which has an expected approximation ratio of  $1/2$ . A vertex will be unsatisfied with probability  $2^{-3}$ . From the linearity of expectation we have that the greedy procedure will improve the cut by at least  $\frac{3}{4.8}|V|$ . Since  $|V| = \frac{2}{3}|E|$ , we achieve approximation ratio at least  $\frac{1}{2} + \frac{1}{16} = 0.5625$  in expectation.

The algorithm  $\mathcal{A}^+$  proceeds by using the variational family of states defined by the algorithm  $\mathcal{A}$  to obtain an approximate cut, but this step is modified or ‘twisted’, as discussed below. The algorithm  $\mathcal{A}^+$  then attempts to enlarge the cut size of the obtained cut by applying a classical post-processing procedure: we perform either the FKL post-processing procedure (obtaining an algorithm FKL- $\mathcal{A}^+$ ) or the HLZ post-processing procedure (giving an algorithm HLZ- $\mathcal{A}^+$ ).

Let us now describe the sense in which  $\mathcal{A}^+$  is a ‘twisted’ form of  $\mathcal{A}$  and not merely a hybrid algorithm augmented by a subsequent classical post-processing step. This terminology stems from the fact that in the quantum subroutine of the algorithm, the variational parameters (angles) are not optimized with respect to the original problem Hamiltonian  $H_G$ . Instead, one can express the expected cut size produced by measuring a state  $\Psi(\theta)$  and using classical post-processing by the expectation value of a modified Hamiltonian  $H_G^+$  (for both FKL and HLZ) in the variational state  $\Psi(\theta)$ . The twisted algorithm  $\mathcal{A}^+$  thus optimizes the angle  $\theta$  with respect to the modified Hamiltonian  $H_G^+$ . Importantly, this does not change the ansatz/variational family of states used. This allows us to make a fair comparison (in terms of quantum resources and, especially, the number of variational parameters) to the original algorithm  $\mathcal{A}$ .

**Improved hybrid algorithms.** The modified algorithm  $\mathcal{A}^+$  requires a set of quantum operations that are comparable (in number and complexity) to that of  $\mathcal{A}$ . In particular, it involves preparing the states  $\{\Psi(\theta)\}_{\theta \in \Theta}$ . In addition,  $\mathcal{A}^+$  uses extra local measurements because the hybrid optimization step is modified: the energy to be optimized is given by a modified problem Hamiltonian  $H_G^+$  rather than the MaxCut-problem Hamiltonian  $H_G$  associated with the considered graph  $G$ . The modified Hamiltonian  $H_G^+$  is either a three- or four-local Hamiltonian and (as  $H_G$ ) diagonal in the computational basis. In particular, this means that measurements of up to 4 qubits at a time in the computational basis are sufficient to determine the (expected) cost function. We note that while this can also be achieved by measuring each qubit in the computational basis and taking appropriate marginals, locality properties can be exploited at the optimization stage, see e.g. [15].

By construction, the algorithms  $\mathcal{A}$  and  $\mathcal{A}^+$  achieve (expected) cut sizes (for any fixed instance  $G$ ) related by the inequalities

$$\mathbb{E} [\text{cutsizes}(\mathcal{A}(G))] \leq \mathbb{E} [\text{cutsizes}(\mathcal{A}^+(G))] \quad (1)$$

for any (bare) hybrid algorithm  $\mathcal{A}$ , assuming that the optimal parameters are found in the optimization step. Indeed, (1) follows because, denoting with

$$\theta_* = \arg \max_{\theta} \langle \Psi(\theta) | H_G | \Psi(\theta) \rangle$$

the optimal parameters for the Hamiltonian  $H_G$ , we have by definition of the algorithms that

$$\begin{aligned} \mathbb{E} [\text{cutsizes}(\mathcal{A}(G))] &= \langle \Psi(\theta_*) | H_G | \Psi(\theta_*) \rangle \\ \mathbb{E} [\text{cutsizes}(\mathcal{A}^+(G))] &= \max_{\theta} \langle \Psi(\theta) | H_G^+ | \Psi(\theta) \rangle, \end{aligned} \quad (2)$$

and

$$H_G^+ = H_G + \Delta_G,$$

where  $\Delta_G$  is a sum of non-negative local operators. These considerations apply to any bare hybrid algorithm  $\mathcal{A}$ .

**Lower bounds on approximation ratios.** We specialize our considerations to QAOA<sub>*p*</sub> and establish lower bounds on the approximation ratio for bare and twisted QAOA, i.e., we consider the algorithms QAOA<sub>*p*</sub> and QAOA<sub>*p*</sub><sup>+</sup>. Specifically, we consider low values of *p* for three-regular graphs, triangle-free three-regular graphs and high girth three-regular graphs. We denote the expected approximation ratio achieved by an algorithm  $\mathcal{A}$  on a graph *G* with maximum cut size MC(*G*) by

$$\alpha_G(\mathcal{A}) := \text{MC}(G)^{-1} \cdot \mathbb{E}[\text{cutsize}(\mathcal{A}(G))].$$

In the following, we will refer to the expected approximation ratio achieved by an algorithm  $\mathcal{A}$  simply as the approximation of  $\mathcal{A}$  (omitting the term ‘expected’) unless specified otherwise. In the case of  $\mathcal{A} = \text{QAOA}_p$ ,  $\mathbb{E}[\text{cutsize}(\mathcal{A}(G))]$  is defined as in (2), but with the level-*p* QAOA trial function  $\Psi_G(\beta, \gamma)$ ,  $\beta, \gamma \in [0, 2\pi]^p$  instead of  $\Psi(\theta)$ .

Our results are summarized in figure 1, which gives our lower bounds on the approximation ratio for each of these methods. For comparison, we also state the following known bounds on bare QAOA for any three-regular graph *G*,

$$\begin{aligned} \alpha_G(\text{QAOA}_1) &\geq 0.6924 && \text{established in [8]} \\ \alpha_G(\text{QAOA}_2) &\geq 0.7559 && \text{conjectured in [8], established in [18]} \\ \alpha_G(\text{QAOA}_3) &\geq 0.79239 && \text{conjectured in [18]}. \end{aligned}$$

Also shown in figure 1 are the guaranteed approximation ratios of the best-known classical algorithms: this includes the Goemans–Williamson algorithm (GW) for general graphs (which is optimal when assuming the unique games conjecture [13]) which achieves

$$\alpha_G(\text{GW}) \geq 0.8785 \quad \text{for any graph } G \text{ (see [10]).}$$

For three-regular graphs, the best efficient classical algorithms are the algorithm by Feige *et al* [9] which relies on a semidefinite program whose solution is then improved by a simple greedy post-processing technique, and a refinement of this technique by Halperin *et al* [11]. They achieve

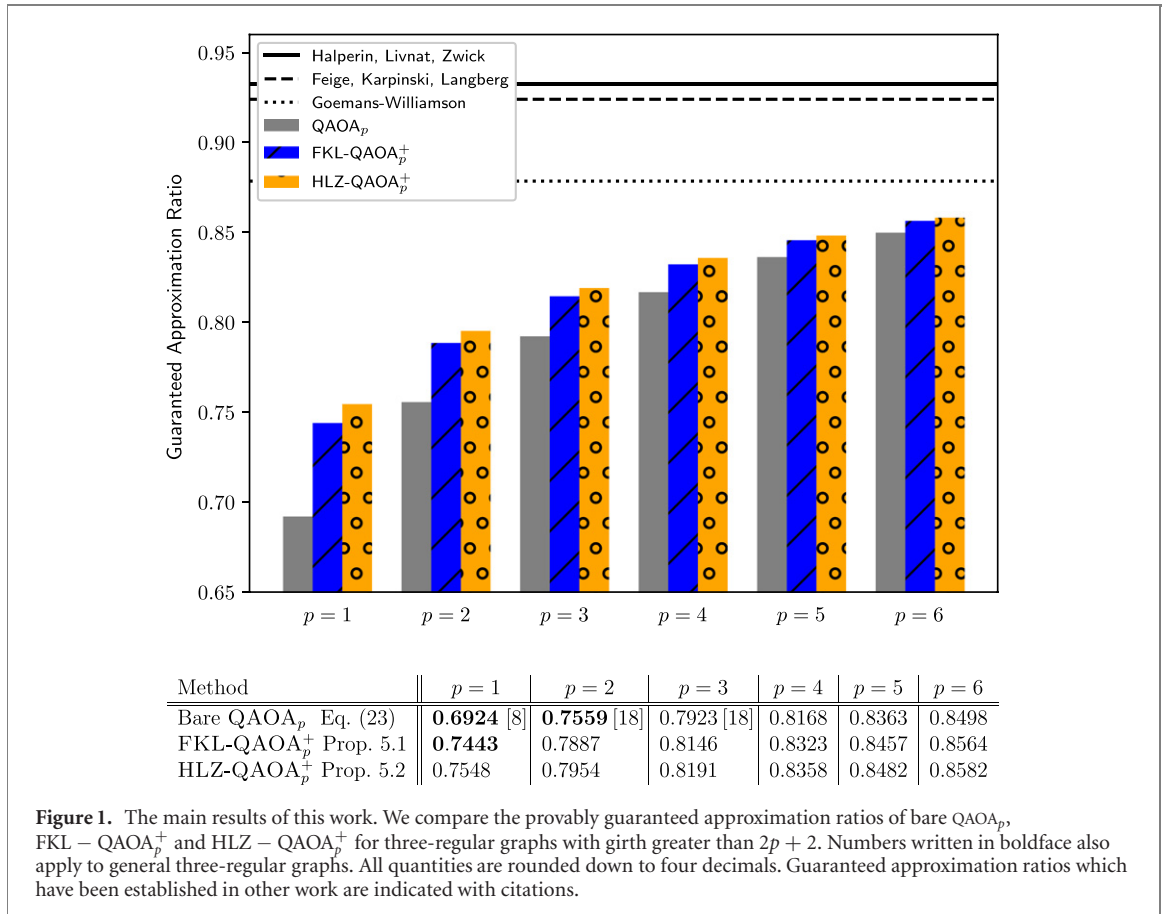
$$\begin{aligned} \alpha_G(\text{FKL}) &\geq 0.924 && \text{for any three-regular graph } G \text{ see [9]} \\ \alpha_G(\text{HLZ}) &\geq 0.9326 && \text{see [11]}. \end{aligned}$$

According to the table given in figure 1, the established lower bounds on the expected approximation ratios for twisted versions of QAOA at level  $p = 1, \dots, 5$  are comparable to the lower bounds on QAOA<sub>*p*+1</sub> at the higher level  $p + 1$ . This suggests that by using these twisted versions, the level *p* can be reduced by one while roughly maintaining the approximation ratio. We emphasize, however, that this conclusion can only be drawn when it is known that the corresponding bound on QAOA<sub>*p*+1</sub> is tight.

Let us conclude by mentioning a few open problems. One potential avenue to obtaining improved approximation ratios with hybrid algorithms is to use a different variational family of ansatz states. Here our work gives clear guidance when this is combined with classical post-processing: for a graph *G*, the energy of a modified cost function Hamiltonian  $H_G^+ = H_G + \Delta_G$  should be optimized instead of that of  $H_G$ . In particular, since  $\Delta_G$  is a sum of three-local terms in the case of FKL and a sum of four-local terms in the case of HLZ, this motivates introducing new terms (e.g., proportional to these terms) in the ansatz. Such a modification of the algorithm is superficially related to the fact that the classical (randomized rounding-based) algorithms of [9, 11] also use additional (three-variable) constraints in the semidefinite program (SDP) compared to the Goemans–Williamson algorithm. We note, however, that using different variational ansatz states will require a different accounting of resources (e.g., circuit depth). In contrast, our twisted algorithms use the same circuits to prepare ansatz states as their bare version.

Another promising approach may be to combine warm-starting-type ideas with classical post-processing. Here one could consider algorithms that first solve the SDP underlying the classical algorithms [9, 11], and subsequently prepare a corresponding quantum state. One may hope that—similar to [5]—suitably designed approaches give a guaranteed approximation ratio matching that of these classical algorithms.

Moving beyond combinatorial optimization problems, it is natural to ask if variational quantum algorithms for many-body quantum Hamiltonian problems (e.g., quantum analogues of MaxCut as considered in [1]) can be improved by similar greedy (quantum) post-processing procedures.



### 1.1.1. Outline

In section 2, we review the relevant classical post-processing methods that—in combination with randomized rounding of the solution of certain SDP relaxations—yield the best known efficient classical algorithms for MaxCut on three-regular graphs. In section 3, we review the QAOA and state a few properties relevant to our subsequent analysis. In section 4, we motivate and define the algorithm  $\mathcal{A}^+$  obtained from a hybrid algorithm  $\mathcal{A}$ . Finally, in section 5, we establish our lower bounds on the achieved approximation ratio achieved by the twisted algorithm  $\text{QAOA}^+$ .

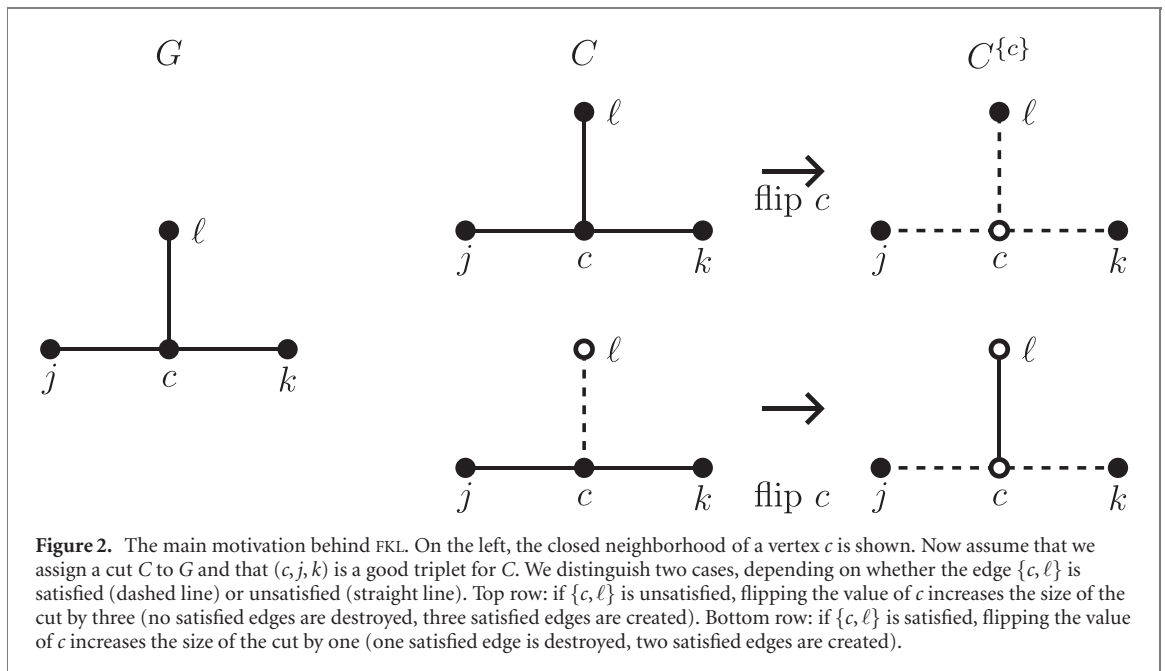
## 2. Classical post-processing methods for MAXCUT

In this section, we describe the two classical post-processing procedures which we build on to define twisted versions of a given hybrid algorithm for the MaxCut problem on three-regular graphs. These post-processing procedures are subroutines of the classical algorithms for MaxCut on bounded degree graphs and graphs with maximum degree 3 by Feige *et al* [9], and Halperin *et al* [11], respectively.

Recall the definition of the MaxCut problem: we are given an (undirected, simple) graph  $G = (V, E)$  and are asked assign two colors to vertices  $C: V \rightarrow \{0, 1\}$ , which we refer to as a cut of  $G$ , that maximizes the number  $\text{cutsize}(C)$  of satisfied edges. Here we say that an edge  $e = \{u, v\}$  is satisfied by  $C$  if and only if  $C(u) \neq C(v)$ . The maximal size  $\text{cutsize}(C)$  of a cut  $C$  of  $G$  is denoted  $\text{MC}(G)$ .

The Goemans–Williamson algorithm [10] for MaxCut proceeds by solving an SDP relaxation [4] of the MaxCut problem, and subsequently uses a randomized hyper-plane rounding to obtain a cut. The algorithms of [9, 11] also proceed by first solving certain SDPs and applying randomized rounding. The obtained candidate cut is then further processed in a greedy manner in order to improve the cut size.

Here we review these post-processing procedures and corresponding performance guarantees. One of their key features is that they can be applied to any candidate cut  $C$  irrespective of whether it is produced e.g., by rounding the solution of an SDP, random guessing, or starting with a fixed cut. This means that they can also be applied to the output of a hybrid algorithm. We emphasize, however, that our modified hybrid algorithms require a modification going beyond simple post-processing of the classical measurement result, see section 4 for details.



Although the guaranteed approximation ratio achieved by HLZ is better than the one achieved by FKL, we investigate both algorithms. The reason for this lies in the locality of the procedures: while FKL considers only the direct neighborhood of a vertex in a single step and is therefore local, HLZ also considers paths and cycles of lengths in the given graph whose lengths might potentially be unbounded and is therefore not necessarily local. We emphasize, however, that the performance of both procedures in the quantum case can be quantified by considering local operators.

Both post-processing procedures take as input a cut  $C$ . They iteratively work towards (ideally) improving the cutsizes by modifying the cut. A single iteration proceeds by identifying a suitable subset  $W \subset V$  of vertices whose assigned color is flipped, i.e., replacing  $C$  by the modified cut

$$C^W(v) := \begin{cases} C(v) & \text{for } v \notin W \\ 1 - C(v) & \text{otherwise} \end{cases}.$$

### 2.1. The Feige–Karpinski–Langberg (FKL) post-processing method

The main idea of this post-processing step is the following observation: if there are three vertices  $c, j, k$  such that one of them (say,  $c$ ) is connected to both the other ones and all three vertices are assigned the same color by the cut  $C$ , then flipping the value at  $c$ , i.e., considering  $C^{\{c\}}$ , will increase the size of the cut, see figure 2.

To formalize this, we assume that the set  $V$  of vertices of the graph  $G = (V, E)$  is ordered. Without loss of generality, set  $V = [n] = \{1, \dots, n\}$ . The following definitions will be central:

Definition 2.1 (triplets).

- (a) A three-tuple  $(c, j, k) \in V^3$  of pairwise distinct vertices with  $j < k$  is called a triplet if  $\{c, j\} \in E$  and  $\{c, k\} \in E$ . We call the vertex  $c$  the central vertex of the triplet. The set of all triplets in  $G$  will be denoted  $T_G$ .
- (b) Let  $C$  be a cut of  $G$  and  $(c, j, k) \in T_G$ . Then  $(c, j, k)$  is called a good triplet for  $C$  if

$$C(c) = C(j) = C(k).$$

The set of all good triplets for  $C$  will be denoted  $\text{Good}_G(C)$ .

- (c) Let  $C$  be a cut of  $G$ ,  $(c, j, k) \in \text{Good}_G(C)$  and  $v \in V$ . We say that  $(c, j, k)$  is destroyed by flipping  $v$  if  $(c, j, k)$  is not a good triplet for the cut  $C^{\{v\}}$ .

We now formulate the post-processing procedure by FKL. While the observations above show that flipping the center of a good triplet  $(c, j, k)$  will increase the cutsizes, we might get even better results by flipping  $j$  or  $k$ . Furthermore, it is in our interest that the flipping does not destroy too many good triplets. Taking all this into account motivates the procedure given in algorithm 1.



**Algorithm 1.** The FKL improvement procedure for three-regular graphs [9].

---

```

1: function FKL(three-regular graph  $G = (V, E)$ , cut  $C$ )
2:    $S \leftarrow \text{Good}_G(C)$ 
3:   while  $S \neq \emptyset$  do
4:      $V_{\text{Good}} \leftarrow$  set of all vertices in  $S$ 
5:      $v \leftarrow \arg \max_{\sigma \in V_{\text{Good}}} \frac{\text{cutsizes}(C^{\{\sigma\}}) - \text{cutsizes}(C)}{|S \setminus \text{Good}_G(C^{\{\sigma\}})|}$ 
6:      $C \leftarrow C^{\{v\}}$ 
7:      $S \leftarrow$  triplets in  $S$  that are good for  $C^{\{v\}}$ 
8:   return  $C$ 

```

---

The following result is proven in [9].

**Lemma 2.2 (lemma 3.2 in [9]).** *Let  $G$  be a three-regular graph and let  $C$  be a cut of  $G$ . Then the cut  $C' = \text{FKL}(G, C)$  satisfies*

$$\text{cutsizes}(C') \geq \text{cutsizes}(C) + \frac{1}{3} |\text{Good}_G(C)|.$$

Let us exemplify this improvement by using two simple examples with a three-regular graph  $G = (V, E)$ . Consider first the trivial constant cut  $C_{\text{const}}$  which assigns the same color to all vertices. The cutsizes of  $C_{\text{const}}$  is 0, hence the approximation ratio vanishes as well, i.e.,

$$\frac{\text{cutsizes}(C_{\text{const}})}{\text{MC}(G)} = 0.$$

Now consider the cut  $C' := \text{FKL}(G, C_{\text{const}})$  obtained by applying the FKL-post-processing procedure to the trivial cut. This cut achieves approximation ratio at least

$$\frac{\text{cutsizes}(C')}{\text{MC}(G)} \geq 2/3.$$

This can be seen as follows: for a constant cut, every triplet is a good triplet and it is easy to see that  $|T_G| = 2|E|$  for a three-regular graph. Lemma 2.2 then implies that the resulting cut  $C$  satisfies  $\text{cutsizes}(C) \geq \frac{2}{3}|E|$  and we obtain the claim with  $\text{MC}(G) \leq |E|$ .

As another example, consider a uniformly random cut  $C_{\text{random}}$  of  $G$ . For such a cut, the expected approximation ratio is

$$\mathbb{E} \left[ \frac{\text{cutsizes}(C_{\text{random}})}{\text{MC}(G)} \right] = 1/2.$$

Let  $C'' := \text{FKL}(G, C_{\text{random}})$  be the result of applying the FKL-procedure to  $C_{\text{random}}$ . Then

$$\mathbb{E} \left[ \frac{\text{cutsizes}(C'')}{\text{MC}(G)} \right] \geq 2/3.$$

To see this, note that the probability of a fixed triplet being good is equal to  $\frac{1}{4}$ . By linearity of expectation, we have  $\mathbb{E}[|\text{Good}_G(C'')|] = \frac{1}{4}|T_G| = \frac{1}{2}|E|$ . Lemma 2.2 then implies that the resulting cut  $C''$  satisfies  $\mathbb{E}[\text{cutsizes}(C'')] \geq \left(\frac{1}{2} + \frac{1}{3} \cdot \frac{1}{2}\right)|E| = \frac{2}{3}|E| \geq \frac{2}{3}\text{MC}(G)$ .

## 2.2. The Halperin–Livnat–Zwick (HLZ) post-processing method

In 2004, Halperin *et al* [11] improved upon the algorithm of [9], giving an algorithm for MaxCut achieving an expected (provable) approximation ratio of at least 0.9326 on graphs with vertex degree at most 3. To the best of our knowledge<sup>3</sup>, this is the best currently known efficient classical algorithm. Although their algorithm works for graphs of maximum degree 3, we will discuss a restricted and thus simpler version for triangle-free three-regular graphs. Unlike the FKL-post-processing this method employs more non-local improvement procedure. The main point here is to illustrate the use of another post-processing method in the construction of twisted hybrid algorithms. We will refer to this procedure simply as HLZ-post-processing.

<sup>3</sup> There is supposedly a slightly improved algorithm in Doror Livnat's M.Sc. Thesis having an approximation ratio 0.9328 [11].

**Algorithm 2.** The HLZ improvement procedure simplified to three-regular triangle free graphs.

```

1: function HLZ(triangle-free three-regular graph  $G = (V, E)$ , cut  $C$ )
2:    $V_3 \leftarrow$  vertices in  $V$  with three unsatisfied edges by cut  $C$ 
3:    $V_2 \leftarrow$  vertices in  $V$  with two unsatisfied edges by cut  $C$ 
4:   while  $V_3 \cup V_2 \neq \emptyset$  do
5:     if  $V_3 \neq \emptyset$  then
6:        $v \leftarrow$  vertex in  $V_3$  with the smallest number of neighbours in  $V_3$ 
7:        $C \leftarrow C^{(v)}$ 
8:     else if  $V_2 \neq \emptyset$  then
9:        $v \leftarrow$  vertex in  $V_2$ 
10:       $\{v_1, \dots, v_k\} \leftarrow$  the longest path or cycle in  $G[V_2]$  containing  $v$ 
11:       $M \leftarrow \{v_i \in \{v_1, \dots, v_k\} | i \text{ is odd}\}$ 
12:       $C \leftarrow C^M$ 
13:       $V_3 \leftarrow$  vertices in  $V$  with three unsatisfied edges by cut  $C$ 
14:       $V_2 \leftarrow$  vertices in  $V$  with two unsatisfied edges by cut  $C$ 
15:   return  $C$ 

```

Given a cut  $C$  of a triangle-free graph  $G$ , this post-processing method proceeds as specified in algorithm 2. Specializing the results of [11] to the triangle-free case considered here gives the following statement:

**Lemma 2.3 (lemma 3.1 in [11]).** *Let  $G$  be a three-regular triangle-free graph,  $C$  be a cut of  $G$  and  $V_2$  and  $V_3$  be the sets of vertices with two and three unsatisfied edges adjacent to them in the cut  $C$ . Then the cut  $C' = \text{HLZ}(G, C)$  satisfies<sup>4</sup>*

$$\text{cutsizesize}(C') \geq \text{cutsizesize}(C) + \frac{2}{5}|V_2| + \frac{17}{15}|V_3|.$$

Again, let us get a feel for the impact of the procedure like we did for FKL in certain simple scenarios, this time for a triangle-free three-regular graph  $G = (V, E)$ . Once again, consider first the trivial constant cut  $C_{\text{const}}$  which assigns the same color to all vertices and therefore has cutsizesize 0, so the approximation ratio is 0 as well. Considering  $C' := \text{HLZ}(G, C_{\text{const}})$ , i.e., the cut obtained by applying the HLZ-post-processing procedure, this cut achieves an approximation ratio of at least

$$\frac{\text{cutsizesize}(C')}{\text{MC}(G)} \geq 0.7555. \quad (3)$$

To see this, note that for a constant cut, all vertices belong to  $V_3 = V$  and none to  $V_2 = \emptyset$ . Lemma 2.3 implies that  $\text{cutsizesize}(C') \geq \frac{17}{15}|V|$  and using that  $|E| = 3/2|V| \geq \text{MC}(G)$ , we obtain

$$\frac{\text{cutsizesize}(C')}{\text{MC}(G)} \geq \frac{\text{cutsizesize}(C')}{|E|} \geq \frac{17 \cdot 2}{15 \cdot 3} \approx 0.7555.$$

As another example, consider a uniformly random cut  $C_{\text{random}}$  of  $G$ . For such a cut, the expected approximation ratio is  $\frac{1}{2}$ , i.e.,  $\mathbb{E}[\text{cutsizesize}(C)] = \frac{1}{2}|E|$ . Considering the cut  $C'' := \text{HLZ}(G, C)$ , the approximation ratio of this cut is

$$\mathbb{E} \left[ \frac{\text{cutsizesize}(C'')}{\text{MC}(G)} \right] \geq 0.6611$$

which can be seen as follows: the probability of a vertex being in  $V_3$  and  $V_2$  are  $2^{-3}$  and  $2^{-2}$ , respectively. By linearity of expectation, we have  $\mathbb{E}[|V_3|] = 2^{-3}|V|$  and  $\mathbb{E}[|V_2|] = 2^{-2}|V|$ . Lemma 2.3 implies that  $\mathbb{E}[\text{cutsizesize}(C'')] \geq \frac{|E|}{2} + \frac{2}{5 \cdot 4}|V| + \frac{17}{15 \cdot 8}|V|$ . Using that  $|V| = \frac{2}{3}|E|$ , we see that the approximation ratio is lower-bounded by  $\frac{1}{2} + \frac{29}{180} \approx 0.6611$  in expectation value.

### 3. Quantum approximate optimization and MaxCut

Here we briefly state the relevant definition for QAOA applied to the MaxCut problem. In section 3.2, we then discuss basic features of QAOA that we exploit to find lower bounds on approximation ratios.

#### 3.1. Definition of the MaxCut Hamiltonian and QAOA<sub>p</sub>

Recall that the MaxCut problem Hamiltonian for a graph  $G = (V, E)$  is given by

$$H_G = \frac{1}{2} \sum_{\{u,v\} \in E} (I - Z_u Z_v) \quad (4)$$

<sup>4</sup> Note that there is a typo in the lemma 3.1 [11].

where a single qubit is associated with each vertex  $u \in V$ . Measurement of a state  $\Psi \in (\mathbb{C}^2)^{\otimes |V|}$  in the computational basis yields a string  $C \in \{0, 1\}^{|V|}$  specifying a cut  $C$  of expected size

$$\langle \Psi | H_G | \Psi \rangle = \mathbb{E}[\text{cutsize}(C)]. \tag{5}$$

The variational family used in QAOA is specified by a natural number  $p$  called the level of QAOA. For a given graph  $G = (V, E)$ , the level- $p$  variational state with parameters  $(\beta, \gamma) \in [0, 2\pi)^p \times [0, 2\pi)^p$  is

$$|\psi_G(\beta, \gamma)\rangle = U_G(\beta, \gamma) |+\rangle^{|V|} \tag{6}$$

where  $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ ,  $|+\rangle^{|V|} := |+\rangle^{\otimes |V|}$  and where

$$U_G(\beta, \gamma) := \prod_{m=1}^p \left[ \exp\left(-i\beta_m \sum_{u \in V} X_u\right) \exp(-i\gamma_m H_G) \right]$$

is the QAOA unitary. In the following, we analyze the performance of twisted algorithms derived from QAOA $_p$ .

### 3.2. Locality and uniformity of QAOA

The analysis of QAOA typically exploits its locality and uniformity, see e.g., [8, 17, 18]. Similar arguments apply to our modified versions of QAOA. Here we state these properties in a form that will be used below to establish lower bounds on the achieved approximation ratios.

**Locality of QAOA.** One of the defining features of this ansatz is its locality: the reduced density operator of  $\psi_G(\beta, \gamma)$  on some subset  $S \subset [n]$  of qubits is uniquely determined by  $(\beta, \gamma)$  and the ‘ $p$ -environment’ of  $S$ , a certain subgraph of  $G$ . For the following analysis, it will be convenient to express this dependence in a more detailed form.

Let  $A$  be a local operator supported on a subset  $\text{supp}(A) \subset [n]$  of qubits. Conjugation of  $A$  by an operator of the form  $\exp(-i\beta_m X_u)$  does not change the support of  $A$  and leaves the operator invariant unless  $u \in \text{supp}(A)$ . Similarly, conjugation of  $A$  by an operator of the form  $\exp(i\gamma_m Z_u Z_v)$  leaves  $A$  invariant unless  $\{u, v\} \cap \text{supp}(A) \neq \emptyset$ , in which case the support generically becomes  $\{u, v\} \cup \text{supp}(A)$ . Applying this reasoning iteratively shows the following: conjugating  $A$  by the QAOA unitary  $U_G(\beta, \gamma)$  is equivalent to conjugation by a cost function unitary  $U_{G^{(p)}[\text{supp}(A)]}(\beta, \gamma)$  associated with a subgraph  $G^{(p)}[\text{supp}(A)]$  of  $G$ . The latter is defined as follows, for any fixed subset  $S \subset V$  vertices corresponding to the support of  $A$ . A length- $\ell$  path starting in  $S$  is a sequence  $(u_0, \dots, u_\ell)$  of vertices such that  $u_0 \in S$  and  $\{u_{j-1}, u_j\} \in E$  for all  $j = 1, \dots, \ell$ . The subgraph  $G^{(p)}[S]$  of  $G$  is the result of taking the union of all paths of length at most  $p$  starting in  $S$ . We call  $G^{(p)}[S]$  the  $p$ -environment of  $S$ . Succinctly, this shows that

$$\langle \psi_G(\beta, \gamma) | A | \psi_G(\beta, \gamma) \rangle = \langle \psi_{G^{(p)}[\text{supp}(A)]}(\beta, \gamma) | A | \psi_{G^{(p)}[\text{supp}(A)]}(\beta, \gamma) \rangle.$$

In other words, to evaluate the expectation of  $A$ , it suffices to consider the QAOA-state associated with the  $p$ -environment of the support of  $A$ .

**Uniformity of QAOA.** For a generic local operator  $A$  with support  $S = \text{supp}(A)$ , the quantity  $\langle \psi_{G^{(p)}[S]}(\beta, \gamma) | A | \psi_{G^{(p)}[S]}(\beta, \gamma) \rangle$  depends on the underlying graph  $G$  only through the  $p$ -environment  $G^{(p)}[S]$  of  $S$  and the subgraph  $G[S]$  of  $G$  induced by  $S$ . In fact, for a fixed induced subgraph  $K := G[S]$ , only the equivalence class of the  $p$ -environment  $G^{(p)}[S]$  matters. Here two graphs  $G_1$  and  $G_2$  (that both contain  $K$  as a subgraph) are called equivalent if and only if they are isomorphic with an isomorphism fixing  $K$ . This property of QAOA is an immediate consequence of its definition.

This motivates considering equivalence classes of  $p$ -environments associated with a graph  $\tilde{G}$ . We denote this set by  $\mathcal{E}^{(p)}(\tilde{G})$  and call this the set of  $p$ -environments of  $\tilde{G}$ . Modulo isomorphisms fixing  $\tilde{G}$ , every element of  $\mathcal{E}^{(p)}(\tilde{G})$  is a graph that appears as a  $p$ -environment  $G^{(p)}[S]$  for a graph  $G$ , where  $S$  is a subset of vertices of  $G$  with the property that the induced subgraph is  $\tilde{G} = G[S]$ . We will use individual representatives of each equivalence class to denote elements of  $\mathcal{E}^{(p)}(\tilde{G})$ . For example, the set  $\mathcal{E}^{(1)}\left(\begin{smallmatrix} \text{---} \\ j \text{---} c \text{---} k \end{smallmatrix}\right)$  is depicted in figure 4 found in appendix A. These observations allow to reorganize expectation values that are uniform. For example,

$$\langle \psi_G(\beta, \gamma) | \left( \sum_{\{u,v\} \in E} Z_u Z_v \right) | \psi_G(\beta, \gamma) \rangle = \sum_{\tilde{G} \in \mathcal{E}^{(p)}\left(\begin{smallmatrix} \text{---} \\ \text{---} \end{smallmatrix}\right)} n_G(\tilde{G}) \langle \psi_{\tilde{G}}(\beta, \gamma) | Z_1 Z_2 | \psi_{\tilde{G}}(\beta, \gamma) \rangle, \tag{7}$$

where  $n_G(\tilde{G})$  is the number of times the  $p$ -environment  $\tilde{G}$  appears in  $G$ .

Of special interest to us will be so-called  $p$ -trees. Given a graph  $\tilde{G}$  and  $p \in \mathbb{N}$ ,  $T^{(p)}(\tilde{G})$  is defined as the sole tree in  $\mathcal{E}^{(p)}(\tilde{G})$ , see figures 6 and 7 in appendix A for examples.

#### 4. Twisted variational hybrid algorithms for MaxCut

In this section, we define our twisted algorithm  $\mathcal{A}^+$  given a hybrid algorithm  $\mathcal{A}$ . We first show in section 4.1 that the effect of classical post-processing can be quantified in terms of the expectation value of a modified problem Hamiltonian. We then give the definition of the twisted algorithm  $\mathcal{A}^+$  in section 4.2.

##### 4.1. Lifting performance guarantees to hybrid algorithms

Lemmas 2.2 and 2.3 provide performance guarantees for the improvement obtained by applying the (classical) FKL- and the HLZ-algorithm to any cut  $C$ . Here we show that these results easily translate to the context of hybrid algorithms.

Concretely, consider a graph  $G = (V, E)$  with  $V = [n]$  and a variational ansatz state  $\Psi \in (\mathbb{C}^2)^{\otimes n}$ . Measuring  $\Psi$  in the computational basis provides a cut  $C \in \{0, 1\}^n$  to which we can apply either the FKL or the HLZ procedure.

Let us first consider the simpler case of FKL, i.e., suppose that  $C' = \text{FKL}(G, C)$  is the cut obtained by applying the FKL-post-processing to the cut  $C$ . To make lemma 2.2 applicable to this setting, we need an operator that accounts for good triplets. Such an operator is

$$N_G := \sum_{(c,j,k) \in T_G} \Pi_{c,j,k}, \quad \text{where } \Pi_{c,j,k} := (|000\rangle\langle 000| + |111\rangle\langle 111|)_{c,j,k}$$

with  $T_G$  denoting the set of triplets in  $G$ . Observe that  $\Pi_{c,j,k}$  is a projector onto the subspace spanned by computational basis states  $|C\rangle$  describing a cut  $C \in \{0, 1\}^n$  such that  $(c, j, k)$  is a good triplet in  $C$ . This implies that the expectation  $\langle \Psi | N_G | \Psi \rangle$  of  $N_G$  in a state  $\Psi$  is equal to the expected number of triplets in a cut  $C$  obtained by measuring  $\Psi$  in the computational basis, i.e.,

$$\langle \Psi | N_G | \Psi \rangle = \sum_{C \in \{0,1\}^n} |\langle C | \Psi \rangle|^2 \cdot |\text{Good}_G(C)| = \mathbb{E} [|\text{Good}_G(C)|]. \tag{8}$$

Correspondingly, we call  $N_G$  the good triplet number operator.

Combining (8) with (5), we obtain the following ‘quantum version’ of lemma 2.2:

**Lemma 4.1.** *Let  $G = (V, E)$  be a three-regular graph with  $V = [n]$  and  $\Psi \in (\mathbb{C}^2)^{\otimes n}$ . Let  $C \in \{0, 1\}^n$  be the result of measuring  $\Psi$  in the computational basis and  $C' := \text{FKL}(G, C)$ . Then*

$$\mathbb{E} [\text{cutsize}(C')] = \langle \Psi | \left( H_G + \frac{1}{3} N_G \right) | \Psi \rangle.$$

This lemma shows that the ‘target Hamiltonian’  $H_G$  should be modified by introducing the improvement operator

$$\Delta_G^{\text{FKL}} := \frac{1}{3} N_G. \tag{9}$$

A similar treatment applies to the HLZ-procedure. Suppose that  $C' = \text{HLZ}(G, C)$  is the cut obtained by applying the HLZ-post-processing to the cut  $C$ . We now want to ‘quantify’ lemma 2.3 and therefore need two operators that account for the number of vertices with two and three unsatisfied edges adjacent to them, respectively. To define these operators, let  $A(c)$  be the ordered three-tuple of neighbors of  $c \in V$  and  $\bar{A}(c)$  denote the closed neighbourhood  $\bar{A}(c) := (c, A(c)_1, A(c)_2, A(c)_3)$ . Then we set

$$M_G^{(2)} = \sum_{c \in V} \Pi_{c, A(c)}^{(2)} \quad \text{and} \quad M_G^{(3)} = \sum_{c \in V} \Pi_{c, A(c)}^{(3)},$$

where

$$\begin{aligned} \Pi_{c, A(c)}^{(2)} &:= \sum_{b \in \{0,1\}} |b\rangle\langle b|_c \otimes P_{A(c)}^{(b)} \quad \text{with } P_{A(c)}^{(b)} := \sum_{\substack{\{x,y,z\} \in \{0,1\}^3, \\ b \oplus x + b \oplus y + b \oplus z = 1}} |xyz\rangle\langle xyz|_{A(c)} \quad \text{and} \\ \Pi_{c, A(c)}^{(3)} &:= (|0000\rangle\langle 0000| + |1111\rangle\langle 1111|)_{\bar{A}(c)}. \end{aligned}$$

**Algorithm 3.** The twisted algorithm  $\text{Post-}\mathcal{A}^+$  where  $\text{Post} \in \{\text{FKL}, \text{HLZ}\}$  and where  $\mathcal{A} = \{|\Psi_G(\theta)\rangle\}_{\theta \in \Theta}$  is a variational algorithm. The measurement result  $C \in \{0, 1\}^n$  obtained in step 3 defines a cut of  $G$ .

---

```

1: function  $\text{Post-}\mathcal{A}^+$  (three-regular graph  $G = (V, E)$  with  $V = [n]$ )
2:   Compute  $\theta_* = \arg \max_{\theta \in \Theta} \langle \Psi_G(\theta) | (H_G + \Delta_G^{\text{Post}}) | \Psi_G(\theta) \rangle$ 
3:   Measure  $\Psi_G(\theta_*)$  in the computational basis getting outcome  $C \in \{0, 1\}^n$ 
4:   Compute  $C' = \text{Post}(G, C)$ 
5:   return  $C'$ 

```

---

Observe that  $P_{A(c)}^{(b)}$  is a projector onto the sum of computational basis states that contain exactly two bits equal to  $b$ . Furthermore,  $\Pi_{c, A(c)}^{(2)}$  is a projector onto the subspace spanned by computational basis states which are associated with exactly two unsatisfied edges adjacent to  $c$ . Similarly,  $\Pi_{c, A(c)}^{(3)}$  is a projector onto the subspace spanned by computational basis states which are associated with exactly three unsatisfied edges adjacent to  $c$ . By abuse of notation, we use  $\Pi_c^{(2)}$  and  $\Pi_c^{(3)}$  whenever the graph is known from the context.

Using the same reasoning as for lemma 4.1, we obtain the following:

**Lemma 4.2.** Let  $G = (V, E)$  be a three-regular triangle-free graph with  $V = [n]$  and  $\Psi \in (\mathbb{C}^2)^{\otimes n}$ . Let  $C \in \{0, 1\}^n$  be the result of measuring  $\Psi$  in the computational basis and  $C' := \text{HLZ}(G, C)$ . Then

$$\mathbb{E}[\text{cutsize}(C')] = \langle \Psi | \left( H_G + \frac{2}{5} M_G^{(2)} + \frac{17}{15} M_G^{(3)} \right) | \Psi \rangle.$$

Therefore,  $H_G$  should be modified by introducing the improvement operator

$$\Delta_G^{\text{HLZ}} := \frac{2}{5} M_G^{(2)} + \frac{17}{15} M_G^{(3)}. \tag{10}$$

#### 4.2. Definition of the twisted algorithm $\mathcal{A}^+$

Here we present our modified variational algorithm  $\mathcal{A}^+$  which we call twisted- $\mathcal{A}$ . We formalize a variational quantum algorithm  $\mathcal{A}$  as follows: it is given by a family of states

$$\mathcal{A} = \{ \Psi_x(\theta) \}_{\theta \in \Theta},$$

where  $x$  is an input to the algorithm, i.e., a problem instance and  $\Theta \subset \mathbb{R}^k$  for some  $k \in \mathbb{N}$ . Once one has chosen  $\theta$ , the state  $\Psi_x(\theta)$  is measured to obtain the output of the algorithm.

In the case of the MaxCut problem, a problem instance is given by a graph  $G$ . A good hybrid algorithm for this problem specifies a variational family  $\{ \Psi_G(\theta) \}_{\theta \in \Theta}$  whose elements can be efficiently prepared (e.g., by a low-depth circuit) and which—ideally—contains elements with large energy (corresponding to the expected cut size) with respect to the MaxCut problem Hamiltonian  $H_G$  (see equation (4)). Given such an algorithm  $\mathcal{A}$ , we obtain a twisted algorithm  $\text{Post-}\mathcal{A}^+$  by the following modifications, where  $\text{Post} \in \{\text{FKL}, \text{HLZ}\}$  denotes the chosen classical post-processing involved (see section 2):

- (a) In the angle optimization step, the modified cost function Hamiltonian  $H_G^+ = H_G + \Delta_G^{\text{Post}}$  is used. Here  $\Delta_G^{\text{FKL}}$  and  $\Delta_G^{\text{HLZ}}$  are the corresponding operators defined in equations (9) and (10), respectively.
- (b) The classical post-processing procedure  $\text{Post}$  is applied to the measurement result obtained by measuring the optimal state.

Algorithm 3 shows the general procedure.

### 5. Lower bounds on approximation ratios of QAOA<sup>+</sup>

Here we analyze the twisted versions of QAOA in detail. For a graph  $G$  and  $p \in \mathbb{N}$ , let  $H_G$  be the Hamiltonian (4) and  $\psi_G(\beta, \gamma)$  the level- $p$  trial wavefunction defined by (6). The twisted algorithms  $\text{FKL} - \text{QAOA}_p^+$  and  $\text{HLZ} - \text{QAOA}_p^+$  proceed as described in algorithm 4. We prove lower bounds on the approximation ratios  $\alpha_G(\text{FKL} - \text{QAOA}_p^+)$  and  $\alpha_G(\text{HLZ} - \text{QAOA}_p^+)$  for certain families of three-regular graphs  $G$ .

A remark on the proof technique is in order here: while we rely on numerical gradient descent to determine good candidate parameters, these are used to optimize our lower bounds only. In particular, the validity of the established bounds is independent of the correctness of these numerical methods. This is especially important because we consider high-dimensional optimization problems and gradient descent may or may not converge.

**Algorithm 4.** The twisted algorithm Post-QAOA<sub>p</sub> for Post ∈ {FKL, HLZ}.

```

1: function Post-QAOAp(three-regular graph  $G = (V, E)$  with  $V = [n]$ )
2:   Compute  $(\beta_*, \gamma_*) = \arg \max_{(\beta, \gamma) \in [0, 2\pi]^p \times [0, 2\pi]^p} \langle \psi_G(\beta, \gamma) | (H_G + \Delta_G^{\text{Post}}) | \psi_G(\beta, \gamma) \rangle$ 
3:   Measure  $\psi_G(\beta_*, \gamma_*)$  in the computational basis getting outcome  $C \in \{0, 1\}^n$ 
4:   Compute  $C' = \text{Post}(G, C)$ 
5:   return  $C'$ 
    
```

**5.1. Approximation ratios of FKL-QAOA<sup>+</sup> for three-regular graphs**

We denote the girth of a graph  $G$ , i.e., the size of the smallest cycle in  $G$ , by  $g(G)$ . We present two kinds of results: for FKL – QAOA<sub>1</sub><sup>+</sup>, we give a bound applicable to all three-regular graphs. For higher levels  $p$ , we give bounds applicable to three-regular graphs with high girth.

**Proposition 5.1.** *Let  $G$  be a three-regular graph. Then*

- (a)  $\alpha_G(\text{FKL} - \text{QAOA}_1^+) \geq 0.7443$ .
- (b) *If  $g(G) \geq 7$ , then  $\alpha_G(\text{FKL} - \text{QAOA}_2^+) \geq 0.7887$ .*
- (c) *If  $g(G) \geq 9$ , then  $\alpha_G(\text{FKL} - \text{QAOA}_3^+) \geq 0.8146$ .*
- (d) *If  $g(G) \geq 11$ , then  $\alpha_G(\text{FKL} - \text{QAOA}_4^+) \geq 0.8323$ .*
- (e) *If  $g(G) \geq 13$ , then  $\alpha_G(\text{FKL} - \text{QAOA}_5^+) \geq 0.8457$ .*
- (f) *If  $g(G) \geq 15$ , then  $\alpha_G(\text{FKL} - \text{QAOA}_6^+) \geq 0.8564$ .*

**Proof.**

- (a) For brevity, let us write  $\psi_G(\theta)$  for the QAOA<sub>1</sub> state with parameters  $\theta = (\beta, \gamma) \in [0, 2\pi]^2$ . Recall from lemma 4.1 that the expected approximation ratio obtained from such a state using the FKL-post-processing procedure is given by

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{FKL}}) | \psi_G(\theta) \rangle}{\text{MC}(G)}. \tag{11}$$

We follow and simplify the approach of [8, 18] and bound the ratio (11) in terms of its local contributions.

We first rearrange and express the numerator of (11) as a sum over triplets. Notice that since the graph is three-regular, any edge lies in exactly four triplets. Hence

$$H_G + \Delta_G^{\text{FKL}} = \sum_{(c,j,k) \in T_G} T_{(c,j,k)} \tag{12}$$

where  $T_{(c,j,k)}$  is the triplet operator defined as

$$T_{(c,j,k)} := \frac{H^{c,j} + H^{c,k}}{4} + \frac{1}{3} \Pi_{c,j,k} \quad \text{for } (c, j, k) \in T_G$$

and where  $H^{a,b} := \frac{1}{2}(I - Z_a Z_b)$  is term in the MaxCut-problem Hamiltonian  $H_G$  associated with the edge  $\{a, b\}$ .

Next consider the denominator in the expression (11), i.e., the maximum size  $\text{MC}(G)$  of a cut. We can bound this term by the expression

$$\text{MC}(G) \leq |E| - |\Delta(G)| - |\diamond(G)|, \tag{13}$$

where  $\Delta(G)$  is the set of isolated triangles (triangles that do not share an edge with another triangle) in  $G$  and  $\diamond(G)$  is the set of crossed squares (consisting of two triangles sharing an edge). Inequality (13) follows immediately from the expression that in any cut of  $G$ , there is at least one unsatisfied (i.e., ‘uncut’) edge in each isolated triangle because of frustration. Similarly, there is at least one unsatisfied edge in each crossed square. We note that the bound (13) applies to any three-regular graph  $G$  with more than four vertices because in these graphs, any triangle is either isolated or part of a crossed squared. (Observe that for the remaining graph, the complete graph  $G = K_4$  on four vertices, we have  $\text{MC}(K_4) = 4$ , and equation (13) does not hold for this graph. In our argument, we will replace equation (13) by the relaxed equation (14) below which applies also to  $K_4$ .)

We can bound  $MC(G)$  further starting from (13) by expressing the right-hand side as a sum over edges. Since every isolated triangle has three edges, we can express the number of isolated triangles as

$$|\Delta(G)| = \frac{1}{3} \sum_{e \in E} \delta_{\Delta(G)}(e),$$

where  $\delta_{\Delta(G)}(e)$  is 1 if the edge  $e$  is part of an isolated triangle in the graph  $G$  and 0 otherwise. Similarly, we have

$$|\diamond(G)| = \frac{1}{5} \sum_{e \in E} \delta_{\diamond(G)}(e)$$

for crossed squares, where  $\delta_{\diamond(G)}(e)$  is 1 if the edge  $e$  is part of a crossed square in the graph  $G$  and 0 otherwise.

To establish our bound, we only consider the one-environment of each edge  $e \in E$ , i.e.,  $G^{(1)}[e]$ . For an edge  $e \in E$  which belongs to a triangle, the one-environment  $G^{(1)}[e]$  is not necessarily sufficient to distinguish whether the triangle is isolated or belongs to a crossed square: for example, this is the case for an edge  $e$  that belongs to a crossed square but is not shared by both triangles. The fraction of uncut edges (in any cut) is  $1/3$  for an isolated triangle, and  $1/5$  for a crossed square. Using the smaller of these two contributions per edge, i.e., pretending that each triangle is in a crossed square, yields the bound

$$MC(G) \leq \sum_{e \in E} \left(1 - \frac{1}{5} \delta_{\Delta(G)}(e)\right). \tag{14}$$

Here  $\delta_{\Delta(G)}(e)$  indicates whether the edge  $e$  is part of a triangle, i.e.,  $\delta_{\Delta(G)}(e)$  equals 1 whenever the edge  $e$  is part of a triangle in graph  $G$  and 0 otherwise. Notice that  $\delta_{\Delta(G)}(e) = \delta_{\Delta(G^{(1)}[e])}(e)$ , therefore it is enough to examine the one-environments of edges to obtain the bound (the possible environments are showcased in figure 3 in the appendix A). We note that while we have excluded  $G = K_4$  in the proof of inequality (14), it is easy to check directly that this graph also satisfies (14).

Expression (14) motivates defining the local averaged MaxCut fraction of an edge  $e$  in  $G$  as

$$L_e^G := 1 - \frac{1}{5} \delta_{\Delta(G)}(e).$$

Using that every edge appears in four triplets, we can reexpress the upper bound (14) as

$$\begin{aligned} MC(G) &\leq \frac{1}{4} \sum_{(c,j,k) \in T_G} (L_{\{c,j\}}^G + L_{\{c,k\}}^G) \\ &= \sum_{(c,j,k) \in T_G} L_{(c,j,k)}^G, \end{aligned} \tag{15}$$

where

$$L_{(c,j,k)}^G := \frac{1}{4} (L_{\{c,j\}}^G + L_{\{c,k\}}^G)$$

denotes the local averaged MaxCut fraction of a triplet  $(c, j, k) \in T_G$ .

Inserting the upper bound (15) on  $MC(G)$  and expression (12) into (11) gives

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{FKL}}) | \psi_G(\theta) \rangle}{MC(G)} \geq \frac{\sum_{(c,j,k) \in T_G} \langle \psi_G(\theta) | T_{(c,j,k)} | \psi_G(\theta) \rangle}{\sum_{(c,j,k) \in T_G} L_{(c,j,k)}^G}. \tag{16}$$

Recall that for any triplet  $(c, j, k) \in T_G$ , the expectation value  $\langle \psi_G(\theta) | T_{(c,j,k)} | \psi_G(\theta) \rangle$  is equal to the local expectation  $\langle \psi_{\tilde{G}}(\theta) | T_{(c,j,k)} | \psi_{\tilde{G}}(\theta) \rangle$ , where  $\tilde{G}$  is the (appropriate) graph environment of the triplet. By its definition as a local quantity, the combinatorial quantity  $L_{(c,j,k)}^G = L_{(c,j,k)}^{\tilde{G}}$  also depends only on the corresponding graph environment. The set of equivalence classes

$\mathcal{E}^{(1)}\left(\overline{j \text{---} c \text{---} k}\right) = \{G_r\}_{r=1}^{11}$  of possible graph environments consists of 11 (equivalence classes of) graphs, see figure 4 in appendix A. Denoting—as in (7)—by  $n_G(G_r)$  the number of times the

environment  $G_r$  appears in  $G$ , we can restate (16) as

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{FKL}}) | \psi_G(\theta) \rangle}{\text{MC}(G)} \geq \frac{\sum_{r=1}^{11} n_G(G_r) \langle \psi_{G_r}(\theta) | T_{(c,j,k)} | \psi_{G_r}(\theta) \rangle}{\sum_{r=1}^{11} n_G(G_r) L_{(c,j,k)}^{G_r}}. \tag{17}$$

Equation (17) is valid for any choice of  $\theta \in [0, 2\pi)^2$ . Suppose now that we have found some angles  $\bar{\theta} \in [0, 2\pi)^2$  such that

$$\frac{\langle \psi_{G_s}(\bar{\theta}) | T_{(c,j,k)} | \psi_{G_s}(\bar{\theta}) \rangle}{L_{(c,j,k)}^{G_s}} \geq \frac{\langle \psi_{G_1}(\bar{\theta}) | T_{(c,j,k)} | \psi_{G_1}(\bar{\theta}) \rangle}{L_{(c,j,k)}^{G_1}} \quad \text{for all } s = 2, \dots, 11. \tag{18}$$

An example of such a pair is

$$\bar{\theta} = (\bar{\beta}, \bar{\gamma}) = (1.130\,565, 5.667\,705) \tag{19}$$

as can be verified by straightforward computation. The median inequality  $\frac{a+b}{c+d} \geq \min\{\frac{a}{c}, \frac{b}{d}\}$  implies (inductively) that

$$\frac{\sum_{r=1}^{11} n_r t_r}{\sum_{r=1}^{11} n_r \ell_r} \geq \min_{r=1, \dots, 11} \frac{t_r}{\ell_r}$$

for any integers  $\{n_j\}_{j=1}^{11} \subset \mathbb{N}_0$  and non-negative scalars  $\{t_r\}_{r=1}^{11}, \{\ell_r\}_{r=1}^{11}$ . Combining this with (18), we conclude that

$$\frac{\sum_{r=1}^{11} n_G(G_r) \langle \psi_{G_r}(\theta) | T_{(c,j,k)} | \psi_{G_r}(\theta) \rangle}{\sum_{r=1}^{11} n_G(G_r) L_{(c,j,k)}^{G_r}} \geq \frac{\langle \psi_{G_1}(\bar{\theta}) | T_{(c,j,k)} | \psi_{G_1}(\bar{\theta}) \rangle}{L_{(c,j,k)}^{G_1}} \geq 0.7443. \tag{20}$$

From (17) and (20) we obtain

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{FKL}}) | \psi_G(\theta) \rangle}{\text{MC}(G)} \geq 0.7443$$

and the claim follows by taking the maximum over  $\theta \in [0, 2\pi)^2$ .

Let us briefly elaborate on the choice (19) of parameters  $\bar{\theta}$  in this proof. By direct computation, we numerically observe that the quantity  $\max_{\theta \in [0, 2\pi)^2} \frac{\langle \psi_{G_r}(\theta) | T_{(c,j,k)} | \psi_{G_r}(\theta) \rangle}{L_{(c,j,k)}^{G_r}}$  is minimal for  $r = 1$ . The parameters  $\bar{\theta} \in [0, 2\pi)^2$  in equation (19) are the numerically obtained angles achieving the maximum for  $r = 1$ . We note that their only required feature in our argument is property (18). This can be verified immediately. A proof that these values  $\bar{\theta}$  indeed correspond to some maximum is not required.

- (b)–(f) Let  $\psi_G(\theta)$  for  $\theta \in [0, 2\pi)^{2p}$  be the QAOA<sub>p</sub>-wave function. We again consider the expected approximation ratio given by the expression ratio (11). We can use the trivial lower bound  $\text{MC}(G) \leq |E|$  on the size of the maximum cut, giving

$$\alpha_G(\text{FKL} - \text{QAOA}_p^+) \geq |E|^{-1} \cdot \langle \psi_G(\theta) | (H_G + \Delta_G^{\text{FKL}}) | \psi_G(\theta) \rangle \tag{21}$$

for any choice of  $\theta \in [0, 2\pi)^{2p}$ . The assumptions on the girth can be expressed as  $g(G) > 2p + 2$  for  $p = 2, 3, 4, 5, 6$ , i.e., the level of QAOA. For such high-girth graphs, all relevant graph environments of an arbitrary triplet in  $G$  are isomorphic to the tree  $T^{(p)}(\frac{\text{---}}{j \ c \ k})$ , see figure 6 in appendix A. Therefore, using (12), the bound (21) becomes

$$\alpha_G(\text{FKL-QAOA}_p^+) \geq 2 \langle \psi_{T^{(p)}(\frac{\text{---}}{j \ c \ k})}(\theta) | T_{(c,j,k)} | \psi_{T^{(p)}(\frac{\text{---}}{j \ c \ k})}(\theta) \rangle \tag{22}$$

for any choice of  $\theta \in [0, 2\pi)^{2p}$ . We can evaluate the right-hand side of this inequality using a tensor network algorithm and gradient descent to maximize the angles. In particular, in each of the cases (b)–(f) we found a set of angles  $\theta$  such that the right-hand side of (22) is equal to the value stated in the proposition. These angles are listed in figure 8. This completes the proof.

For sake of comparison, we also obtained the guaranteed approximation ratios of bare QAOA for  $p = 4, 5$ , and 6 for high girth graphs. These were computed in a similar fashion as explained at the end of the proof of proposition 5.1:

$$\alpha_G(\text{QAOA}_p) \geq \langle \psi_{T^{(p)}(\frac{\text{---}}{1 \ 2})}(\theta) | 2^{-1}(I - Z_1 Z_2) | \psi_{T^{(p)}(\frac{\text{---}}{1 \ 2})}(\theta) \rangle \tag{23}$$

The witness angles proving the lower bounds are listed in figure 8.



### 5.2. Approximation ratios of HLZ-QAOA<sup>+</sup> for three-regular graphs

**Proposition 5.2.** *Let  $G = (V, E)$  be a three-regular graph. Then*


- (a) *If  $G$  is triangle-free (i.e.  $g(G) \geq 4$ ), then  $\alpha_G(\text{HLZ} - \text{QAOA}_1^+) \geq 0.7548$ .*
- (b) *If  $g(G) \geq 7$ , then  $\alpha_G(\text{HLZ} - \text{QAOA}_2^+) \geq 0.7954$ .*
- (c) *If  $g(G) \geq 9$ , then  $\alpha_G(\text{HLZ} - \text{QAOA}_3^+) \geq 0.8191$ .*
- (d) *If  $g(G) \geq 11$ , then  $\alpha_G(\text{HLZ} - \text{QAOA}_4^+) \geq 0.8358$ .*
- (e) *If  $g(G) \geq 13$ , then  $\alpha_G(\text{HLZ} - \text{QAOA}_5^+) \geq 0.8482$ .*
- (f) *If  $g(G) \geq 15$ , then  $\alpha_G(\text{FKL} - \text{QAOA}_6^+) \geq 0.8582$ .*

**Proof.**

- (a) Recall from lemma 4.2 that the expected approximation ratio obtained using the HLZ-post-processing procedure is given by

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{HLZ}}) | \psi_G(\theta) \rangle}{\text{MC}(G)}, \tag{24}$$

where we again use  $\psi_G(\theta)$  for the QAOA<sub>1</sub> state.

We rearrange and express the numerator (24) as a sum over three-star subgraphs, as they are underlying graphs of local terms of the improvement operator  $\Delta_G^{\text{HLZ}}$ . The three-star graph with the central vertex  $c$  has vertices  $\{c, j, k, \ell\}$  and edges  $\{\{c, j\}, \{c, k\}, \{c, \ell\}\}$  and we depict it by . Since the graph  $G$  is three-regular, any edge  $\{a, b\} \in E$  lies in exactly two stars with central vertices  $a$  and  $b$ . Hence

$$H_G + \Delta_G^{\text{HLZ}} = \sum_{c \in V} S_c, \tag{25}$$

where  $S_c$  is the three-star operator

$$S_c := \frac{H^{c,j} + H^{c,k} + H^{c,\ell}}{2} + \frac{2}{5} \Pi_c^{(2)} + \frac{17}{15} \Pi_c^{(3)} \quad \text{for } c \in V,$$

$(j, k, \ell)$  is the ordered neighbourhood of  $c$  in  $G$  and  $H^{a,b}$  is again the MaxCut term on edge  $\{a, b\}$ .

Inserting the trivial upper bound on  $\text{MC}(G) \leq |E|$  and (25) into (24) gives:

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{HLZ}}) | \psi_G(\theta) \rangle}{\text{MC}(G)} \geq \frac{\sum_{c \in V} \langle \psi_G(\theta) | S_c | \psi_G(\theta) \rangle}{|E|}. \tag{26}$$

We can restate (26) as a sum over the local expectation values over the graph environments from

the set  $\mathcal{E}^{(1)} \left( \begin{smallmatrix} j \\ c \\ k \quad l \end{smallmatrix} \right) = \{G_r\}_{r=1}^8$  (listed in figure 5 in appendix A):

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{HLZ}}) | \psi_G(\theta) \rangle}{\text{MC}(G)} \geq \frac{\sum_{r=1}^8 n_G(G_r) \langle \psi_{G_r}(\theta) | S_c | \psi_{G_r}(\theta) \rangle}{|E|}, \tag{27}$$

where  $n_G(G_r)$  is number of times the environment  $G_r$  appears in graph  $G$ .

Suppose now that we have found some angles  $\bar{\theta} \in [0, 2\pi)^2$  such that

$$\langle \psi_{G_s}(\bar{\theta}) | S_c | \psi_{G_s}(\bar{\theta}) \rangle \geq \langle \psi_{G_1}(\bar{\theta}) | S_c | \psi_{G_1}(\bar{\theta}) \rangle \quad \text{for all } s = 2, \dots, 8. \tag{28}$$

An example of such a pair is

$$\bar{\theta} = (\bar{\beta}, \bar{\gamma}) = (0.102870, 5.669319)$$

as can be verified by straightforward computation.

We combine (27) with (28) and use the fact that  $\sum_{r=1}^8 n_G(G_r) = |V| = 2/3|E|$  for three-regular graphs:

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{HLZ}}) | \psi_G(\theta) \rangle}{\text{MC}(G)} \geq \frac{2}{3} \langle \psi_{G_1}(\bar{\theta}) | S_c | \psi_{G_1}(\bar{\theta}) \rangle \geq 0.7548$$

and the claim follows.

(b)–(f) We will follow a similar line of reasoning as in (a) and proposition 5.1(b)–(f). The assumptions again guarantee that the considered graphs are of girth greater than  $2p + 2$  with  $p$  being the level of QAOA. For such high-girth graphs, all graph environments of an arbitrary star in  $G$  are isomorphic to  $\tilde{G} = T^{(p)}\left(\begin{array}{c} j \\ k \quad c \quad l \end{array}\right)$ . Therefore,

$$\alpha_G(\text{HLZ} - \text{QAOA}_p^+) \geq \frac{2}{3} \langle \psi_{\tilde{G}}(\theta) | S_c | \psi_{\tilde{G}}(\theta) \rangle,$$

where  $\psi_{\tilde{G}}(\theta)$  for  $\theta \in [0, 2\pi)^{2p}$  be the  $\text{QAOA}_p$ -wave function. We obtain witness angles by numerical optimization (listed in figure 8) and the claim follows.

We note that the proven lower bound proposition 5.2(a) on the approximation ratio  $\alpha_G(\text{HLZ} - \text{QAOA}_1^+)$  of the twisted algorithm  $\text{QAOA}_1$  is below the value 0.7555 resulting from the application of HLZ to a constant partition (see (3)). An improvement over this trivial (classical) algorithm can only be observed starting from level  $p \geq 2$  (cf proposition (5.2)(b)–(f)). This is not surprising given the fact that the QAOA-ansatz is very restricted, especially for small values of  $p$ . In particular, for any angles  $(\beta, \gamma)$ , the QAOA-state  $\psi_G(\beta, \gamma)$  (cf (6)) with the usual cost function Hamiltonian  $H_G$  for MaxCut is different from both the all-zero state  $|0\rangle^{\otimes n}$  and the all-one state  $|1\rangle^{\otimes n}$ . This is the case for any level  $p$  since because of the  $\mathbb{Z}_2$ -symmetry of the ansatz: every state  $\psi_G(\beta, \gamma)$  is an eigenstate of the operator  $X^{\otimes n}$ .

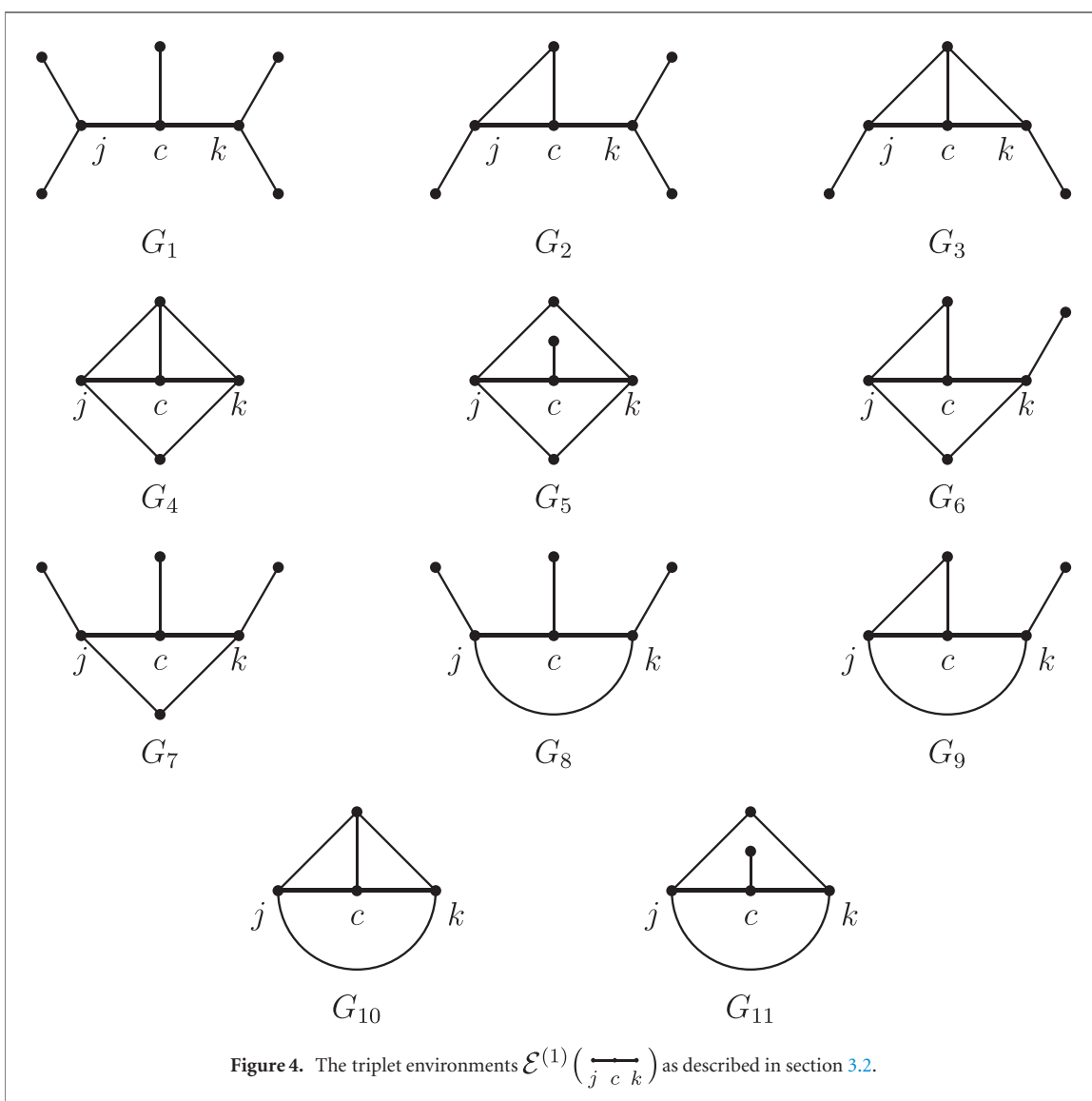
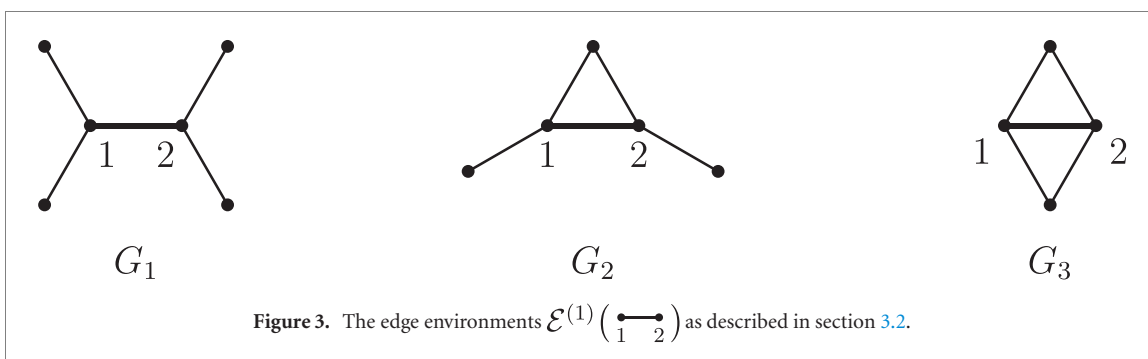
## Acknowledgments

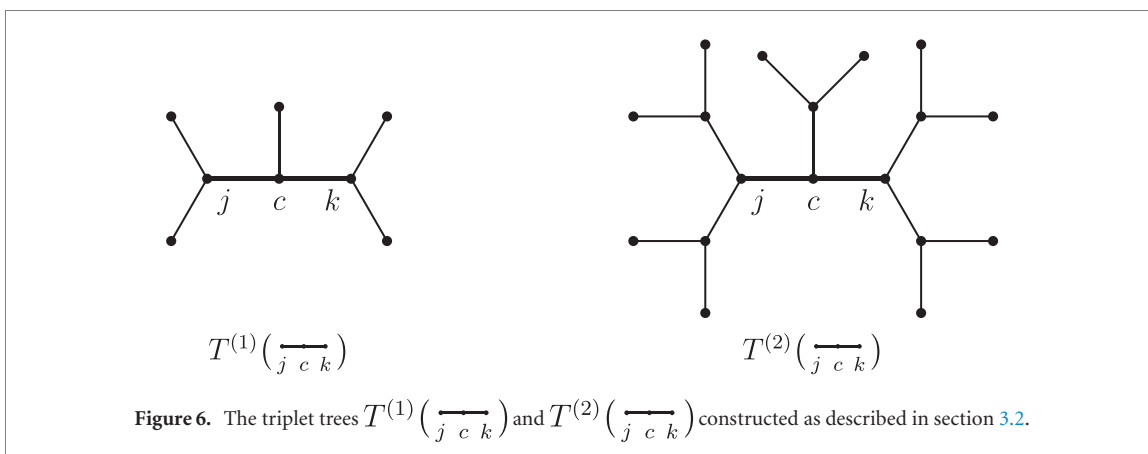
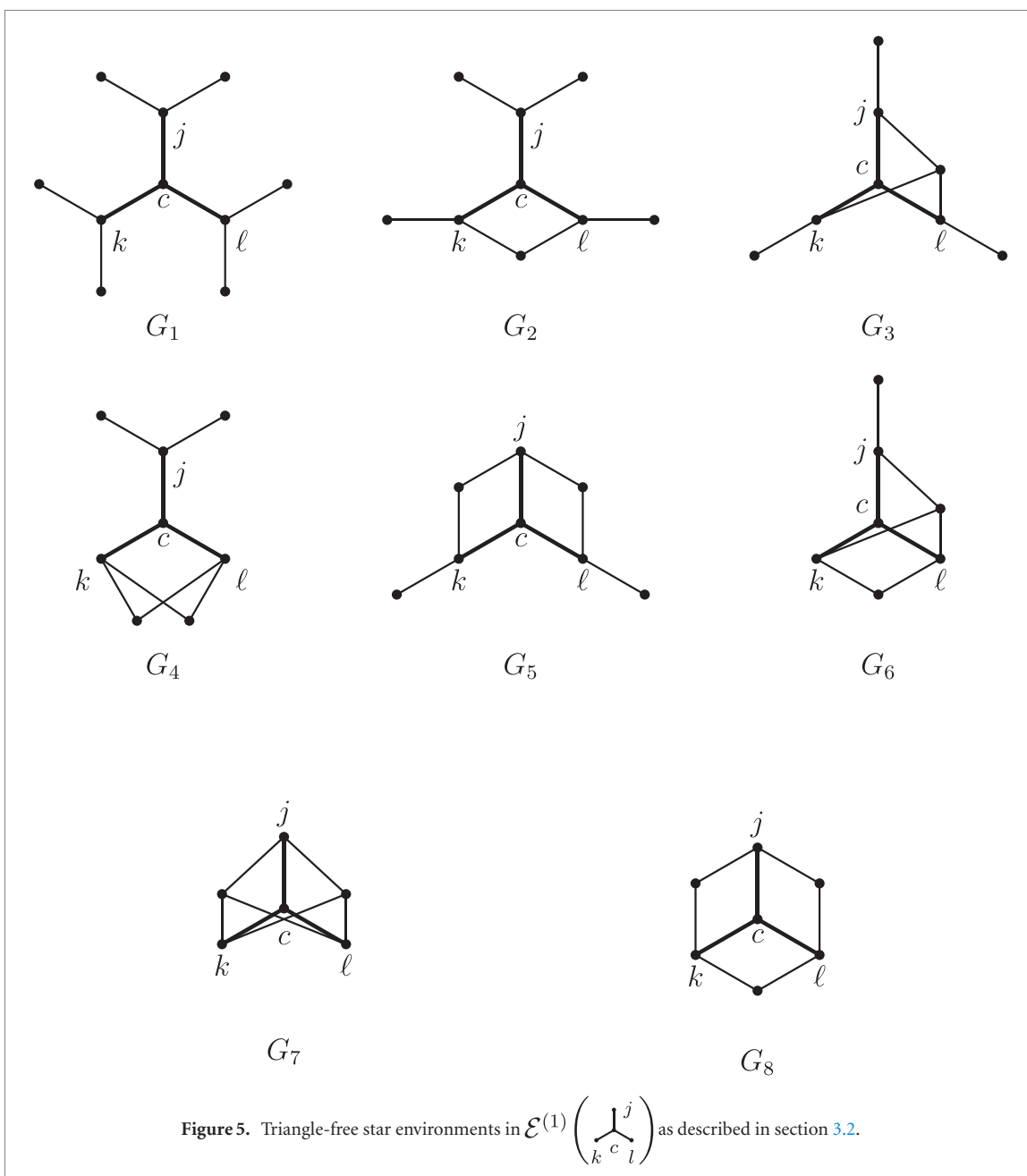
We thank Zahra Baghali Khanian for suggesting the name ‘twisted QAOA’. AK and RK acknowledge support by the Army Research Office under Grant No. W911NF-20-1-0014. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein. RK and LC gratefully acknowledge support by the European Research Council under Grant Agreement No. 101001976 (project EQUIPTNT). LC thanks IBM Zurich for their hospitality.

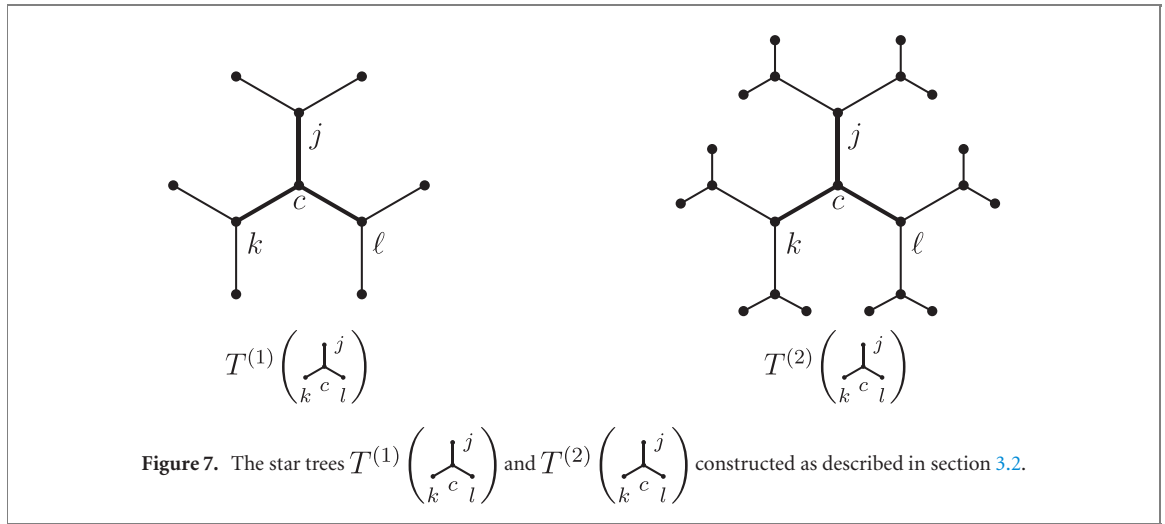
## Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

## Appendix A. Used graph environments







### Appendix B. Witness angles

Method	$\beta$					$\gamma$				
FKL-QAOA <sub>2</sub> <sup>+</sup>	0.99225	3.46308				5.78009	2.25304			
HLZ-QAOA <sub>2</sub> <sup>+</sup>	0.98705	3.47167				5.77664	2.25962			
FKL-QAOA <sub>3</sub> <sup>+</sup>	0.62112	0.48905	0.26477			0.42728	0.79596	0.92620		
HLZ-QAOA <sub>3</sub> <sup>+</sup>	0.62519	0.49754	0.27393			0.42808	0.79569	0.92077		
QAOA <sub>4</sub>	0.59956	0.43434	0.29676	0.15904		0.40875	0.78057	0.98804	0.15691	
FKL-QAOA <sub>4</sub> <sup>+</sup>	0.63219	2.09215	0.42150	0.22286		0.38433	0.72509	0.83266	0.94350	
HLZ-QAOA <sub>4</sub> <sup>+</sup>	0.63516	0.52634	0.43047	0.23058		0.38478	0.72269	0.82767	0.93461	
QAOA <sub>5</sub>	0.63167	0.52253	1.96094	0.27599	0.14930	0.35924	0.70609	0.82209	1.00420	1.15394
FKL-QAOA <sub>5</sub> <sup>+</sup>	0.64008	0.54030	0.45437	0.34000	0.18710	0.35582	0.68736	0.78042	0.87482	0.99556
HLZ-QAOA <sub>5</sub> <sup>+</sup>	0.64349	0.54679	0.46687	0.38838	0.19975	0.35349	0.68144	0.76945	0.85500	0.96997
QAOA <sub>6</sub>	0.63589	0.53443	0.46334	0.35999	0.25858	0.13885	0.33137	0.64558	0.73165	0.83696
FKL-QAOA <sub>6</sub> <sup>+</sup>	0.64369	0.54870	0.47903	0.40547	1.88825	0.16000	0.33434	0.64986	0.73024	0.81681
HLZ-QAOA <sub>6</sub> <sup>+</sup>	0.64622	0.55243	0.48572	0.42258	1.91095	0.17045	0.33265	0.64669	0.72479	0.80326

**Figure 8.** Witness angles  $\beta = \beta_1, \beta_2, \dots$  and  $\gamma = \gamma_1, \gamma_2, \dots$  certifying the approximation ratios of QAOA<sub>p</sub>, FKL – QAOA<sub>p</sub><sup>+</sup> and HLZ – QAOA<sub>p</sub><sup>+</sup> on graphs of girth greater than  $2p + 2$  for  $p \in \{2, \dots, 6\}$ .

### ORCID iDs

Libor Caha <https://orcid.org/0000-0001-7401-8076>

Alexander Kliesch <https://orcid.org/0000-0001-7973-9665>

### References

- [1] Anshu A, Gosset D and Morenz K 2020 Beyond product state approximations for a quantum analogue of max cut *15th Conf. on the Theory of Quantum Computation, Communication and Cryptography (TQC 2020)* vol 158 pp 7:1–7:15
- [2] Bravyi S, Kliesch A, Koenig R and Tang E 2020 Hybrid quantum-classical algorithms for approximate graph coloring *Quantum* **6** 678
- [3] Bravyi S, Kliesch A, Koenig R and Tang E 2020 Obstacles to variational quantum optimization from symmetry protection *Phys. Rev. Lett.* **125** 260505
- [4] Delorme C and Poljak S 1993 Laplacian eigenvalues and the maximum cut problem *Math. Program.* **62** 557–74
- [5] Egger D J, Mareček J and Woerner S 2021 Warm-starting quantum optimization *Quantum* **5** 479
- [6] Farhi E, Gamarnik D and Gutmann S 2020 The quantum approximate optimization algorithm needs to see the whole graph: worst case examples (arXiv:2005.08747)
- [7] Farhi E, Gamarnik D and Gutmann S 2020 The quantum approximate optimization algorithm needs to see the whole graph: a typical case (arXiv:2005.09002)
- [8] Farhi E, Goldstone J and Gutmann S 2014 A quantum approximate optimization algorithm (arXiv:1411.4028)
- [9] Feige U, Karpinski M and Langberg M 2002 Improved approximation of max-cut on graphs of bounded degree *J. Algorithms* **43** 201–19
- [10] Goemans M X and Williamson D P 1995 Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming *J. ACM* **42** 1115–45
- [11] Halperin E, Livnat D and Zwick U 2004 MAX CUT in cubic graphs *J. Algorithms* **53** 169–85

- [12] Kandala A, Mezzacapo A, Temme K, Takita M, Brink M, Chow J M and Gambetta J M 2017 Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets *Nature* **549** 242–6
- [13] Khot S, Kindler G, Mossel E and O'Donnell R 2007 Optimal inapproximability results for MAX-CUT and other 2-variable CSPs? *SIAM J. Comput.* **37** 319–57
- [14] Mossel E, O'Donnell R and Oleszkiewicz K 2010 Noise stability of functions with low influences: invariance and optimality *Ann. Math.* **171** 295–341
- [15] Sack S H, Medina R A, Michailidis A A, Kueng R and Serbyn M 2022 Avoiding barren plateaus using classical shadows (arXiv:2201.08194)
- [16] Tate R, Farhadi M, Herold C, Mohler G and Gupta S 2021 Bridging classical and quantum with SDP initialized warm-starts for QAOA (arXiv:2010.14021)
- [17] Weggemans J R, Urech A, Rausch A, Spreeuw R, Boucherie R, Schreck F, Schoutens K, Minář J and Speelman F 2021 Solving correlation clustering with QAOA and a Rydberg qudit system: a full-stack approach (arXiv:2106.11672)
- [18] Wurtz J and Love P 2021 MaxCut quantum approximate optimization algorithm performance guarantees for  $p > 1$  *Phys. Rev. A* **103** 042612

## Appendix B

### Article as co-author

#### B.1 Obstacles to Variational Quantum Optimization from Symmetry Protection

# Obstacles to Variational Quantum Optimization from Symmetry Protection

Sergey Bravyi, Alexander Kliesch, Robert Koenig, Eugene Tang

---

While already the seminal paper on QAOA [12] provides a non-trivial lower bound on the worst-case approximation ratio achieved by QAOA applied to MAX-CUT on 3-regular graphs, little was known about upper bounds on this quantity in any setting. In particular, it was not rigorously known whether constant-level QAOA might be able to outperform the best classical approximation algorithms in the general case. In Article III) [3], we give a negative answer to this question.

## B.1.1 Main Results

We show that the worst-case approximation ratio achievable by QAOA applied to MAX-CUT, where the level of QAOA is growing at most logarithmically in the size of the considered graphs, is upper bounded by a constant arbitrarily close to  $\frac{5}{6} \approx 0.8333$ . In particular, this implies that the worst-case approximation ratio of constant-level QAOA is worse than the one achieved by the best classical approximation algorithm for MAX-CUT, the Goemans-Williamson algorithm [17], which achieves an approximation ratio of 0.8785. To establish this result, we utilize that the QAOA circuits and the MAX-CUT Hamiltonians commute with the symmetry operator  $X^{\otimes n}$ , where  $n$  is the number of vertices, while the QAOA input state is a +1 eigenstate of this operator; we therefore call the resulting QAOA state  $\mathbb{Z}_2$ -symmetric.

As a second result, we prove a general upper bound on the performance of geometrically local variational algorithms producing  $\mathbb{Z}_2$ -symmetric states by considering the family of cycle graphs. Our results make it clear that all local variational quantum algorithms suffer from restrictions; we therefore introduce a more non-local version of QAOA which we call “recursive QAOA” (RQAOA) which uses QAOA as a subroutine to determine correlations between vertices of a given graph. For the family of cycle graphs, we then show that RQAOA with level 1 produces a maximum cut for any member of the family, strictly surpassing the worst-case performance of constant-level QAOA.

## B.1.2 Individual Contribution

Eugene Tang is the principal author of this article. The idea for working on proving limitations of QAOA first came after discussions with my doctoral advisor Robert Koenig and Eugene Tang who was then visiting our department M5. The concrete idea to show these limitations by exploiting symmetry considerations was conceived while Robert Koenig visited Sergey Bravyi at IBM in New York. I was involved in the scientific work and the write-up of the article.

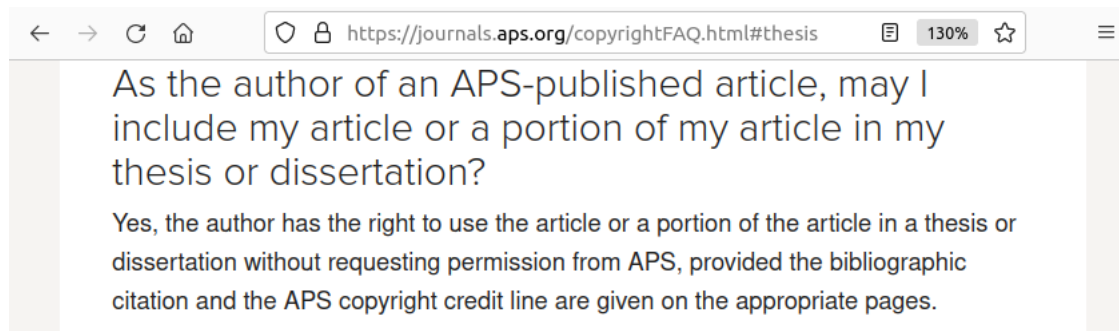



# Permission to include:

Sergey Bravyi, Alexander Kliesch, Robert Koenig, Eugene Tang

“Obstacles to Variational Quantum Optimization from Symmetry Protection”

*Physical Review Letters* 125, 260505 (2020)



**Obstacles to Variational Quantum Optimization from Symmetry Protection**Sergey Bravyi,<sup>1</sup> Alexander Kliesch<sup>2</sup>, Robert Koenig,<sup>3</sup> and Eugene Tang<sup>4</sup><sup>1</sup>*IBM Quantum, IBM T. J. Watson Research Center, Yorktown Heights, New York 10598, USA*<sup>2</sup>*Zentrum Mathematik, Technical University of Munich, 85748 Garching, Germany*<sup>3</sup>*Institute for Advanced Study and Zentrum Mathematik, Technical University of Munich, 85748 Garching, Germany*<sup>4</sup>*Institute for Quantum Information and Matter, Caltech, Pasadena, California 91125, USA* (Received 22 October 2019; revised 16 September 2020; accepted 3 December 2020; published 24 December 2020)

The quantum approximate optimization algorithm (QAOA) employs variational states generated by a parameterized quantum circuit to maximize the expected value of a Hamiltonian encoding a classical cost function. Whether or not the QAOA can outperform classical algorithms in some tasks is an actively debated question. Our work exposes fundamental limitations of the QAOA resulting from the symmetry and the locality of variational states. A surprising consequence of our results is that the classical Goemans-Williamson algorithm outperforms the QAOA for certain instances of MaxCut, at any constant level. To overcome these limitations, we propose a nonlocal version of the QAOA and give numerical evidence that it significantly outperforms the standard QAOA for frustrated Ising models.

DOI: [10.1103/PhysRevLett.125.260505](https://doi.org/10.1103/PhysRevLett.125.260505)

Variational quantum optimization (VQO) has recently received significant attention as a candidate application of near-term quantum processors. The basic proposal is appealingly simple: The output state of a parameterized quantum circuit is used as a variational wave function to minimize the expected energy of a given Hamiltonian [1]. Depending on the envisioned application, the Hamiltonian may govern electronic interactions in a molecule or material of interest [2] or encode a classical cost function whose minimum is to be approximated [3]. Rotation angles that define individual gates in the state preparation circuit serve as variational parameters. These parameters are adjusted via a classical feedback loop that aims to minimize the expected energy.

The central question common to all VQO proposals is whether the chosen variational class of states is expressive enough to provide a good ground state approximation. Let us point out two factors that can limit the expressive power of VQO. First, the state preparation quantum circuit must have a small depth to enable reliable implementation on near-term noisy devices lacking error correction. This means that highly entangled states such as the ground state of Kitaev's toric code [4] may be out of scope for near-term VQO using gate-based devices and low-depth circuits [5,6]. Second, the number of variational parameters in the state preparation circuit must be small enough to enable efficient energy minimization. While this is not a serious concern for proof-of-principle experiments with a handful of qubits, it is anticipated that large-scale VQO with an extensive number of variational parameters may give rise to intractable optimization problems, for example, due to the barren plateau (vanishing gradient) effect [7].

In this Letter, we elaborate on the limitations of VQO and establish formal no-go results for the quantum

approximate optimization algorithm (QAOA) [3]. Recall that the QAOA aims to approximate the maximum of a classical cost function  $C(x)$  that depends on  $n$  binary variables,  $x = (x_1, \dots, x_n)$ . The cost function is encoded into an  $n$ -qubit diagonal Hamiltonian  $C = \sum_{x \in \{0,1\}^n} C(x)|x\rangle\langle x|$ . The QAOA variationally maximizes the expected energy of  $C$  over  $n$ -qubit quantum states of the form [3]

$$|\psi(\beta, \gamma)\rangle = \prod_{a=1}^p e^{i\beta_a B} e^{i\gamma_a C} |+\rangle^n, \quad (1)$$

where  $\beta_a$  and  $\gamma_a$  are variational parameters,  $|+\rangle^n$  is the tensor product of  $n$  single-qubit states  $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$ , and  $B = \sum_{j=1}^n X_j$  is the transverse magnetic field operator. The integer  $p$ , called the QAOA *level*, controls the expressive power of the variational ansatz. Finally, the QAOA outputs a bit string  $x$  obtained by preparing the optimal variational state  $|\psi(\beta, \gamma)\rangle$  and measuring each qubit in the standard basis. The expected value of  $C(x)$  coincides with the variational energy  $\langle \psi(\beta, \gamma) | C | \psi(\beta, \gamma) \rangle$ . The performance of the QAOA is commonly quantified by an *approximation ratio* defined as the ratio between the maximum variational energy and the maximum value of the cost function  $C_{\max} = \max_x C(x)$ .

A paradigmatic test case for the QAOA is the maximum cut (MaxCut) problem [3]. Suppose  $G = (V, E)$  is a graph with a set of  $n$  vertices  $V$  labeled by integers  $j = 1, \dots, n$  and a set of edges  $E$ . Given an  $n$ -bit string  $x$ , let  $\text{cut}(x)$  be the set of edges  $(j, k) \in E$  such that  $x_j \neq x_k$ . The cost function to be maximized is the cut size,  $C(x) = |\text{cut}(x)|$ . The corresponding  $n$ -qubit Hamiltonian is

$$C = \frac{1}{2} \sum_{(j,k) \in E} (I - Z_j Z_k). \quad (2)$$

Here,  $Z_j$  is the Pauli  $Z$  operator applied to a qubit  $j$ , and  $I$  is the identity. The state preparation circuit defined in Eq. (1) has depth  $\approx pD$ , where  $D$  is the maximum vertex degree of the graph  $G$  and  $p$  is the QAOA level [8]. To keep the circuit depth and the number of variational parameters small, below, we focus on the regime when  $p$  and  $D$  are constants or slowly growing functions of  $n$ .

Our first result is an upper bound on the maximum variational energy attained by level- $p$  states. Namely, we show that for any constant  $D \geq 3$  and all large enough  $n$  there exists a degree- $D$  graph  $G$  with  $n$  vertices such that

$$\frac{\langle \psi(\beta, \gamma) | C | \psi(\beta, \gamma) \rangle}{C_{\max}} \leq \frac{5}{6} + \frac{\sqrt{D-1}}{3D} \quad (3)$$

for any  $\beta, \gamma \in \mathbb{R}^p$  as long as  $p < (1/3 \log_2 n - 4)(D+1)^{-1}$ . This result severely limits the performance of the QAOA with any constant level  $p$  independent of  $n$ . Indeed, the right-hand side of Eq. (3) is approximately  $5/6 \approx 0.833$  for large vertex degree  $D$ . For comparison, the best-known classical algorithm for MaxCut due to Goemans and Williamson [9] achieves the approximation ratio  $\approx 0.878$  on an arbitrary graph. Thus, the QAOA with a constant level  $p$  cannot outperform classical algorithms. We note that upper bounds on the QAOA approximation ratio were previously known only for  $p = 1$  [3]. We refer to Ref. [10] for numerical studies of the QAOA applied to MaxCut.

Similar concerns about limitations of the QAOA have previously been voiced by Hastings [11], who showed analytically that certain local classical algorithms match the performance of the level-1 QAOA for Ising-like cost functions with multispin interactions. Hastings also gave numerical evidence for the same phenomenon for MaxCut with  $p = 1$  and argued that this should extend to  $p > 1$  [11].

QAOA states possess a certain symmetry that plays a crucial role in our analysis. Namely, the state  $|\psi(\beta, \gamma)\rangle$  is invariant under a global spin flip:

$$X^{\otimes n} |\psi(\beta, \gamma)\rangle = |\psi(\beta, \gamma)\rangle.$$

Indeed, the Hamiltonians  $B$  and  $C$  commute with the symmetry operator  $X^{\otimes n}$ , while the initial state  $|+\rangle^n$  is a  $+1$  eigenvector of  $X^{\otimes n}$ . More generally, let us say that an  $n$ -qubit state  $|\psi\rangle$  is  $\mathbb{Z}_2$  symmetric if it is a  $+1$  eigenvector of  $X^{\otimes n}$ . Our proof of Eq. (3) combines two observations: (i) The symmetry forces good variational states to be highly entangled, and (ii) low-depth circuits are not capable of preparing highly entangled states.

To elaborate on the role of the  $\mathbb{Z}_2$  symmetry, suppose  $x \in \{0, 1\}^n$  is an optimal cut such that  $C_{\max} = C(x)$ . Let  $\bar{x}$  be the bitwise negation of  $x$ . Note that  $C(\bar{x}) = C(x)$ .

Although the state  $|x\rangle$  has no entanglement whatsoever, its  $\mathbb{Z}_2$ -symmetric version  $(|x\rangle + |\bar{x}\rangle)/\sqrt{2}$  is a highly entangled state, locally equivalent to the  $n$ -qubit Greenberger-Horne-Zeilinger (GHZ) state  $(|0^n\rangle + |1^n\rangle)/\sqrt{2}$ , which cannot be prepared by a low-depth circuit [5].

The fact that symmetry may prevent one from preparing ground states of certain Hamiltonians by low-depth circuits is well known in the theory of topological quantum order under the name *symmetry protection* [12–14]. The bound Eq. (3) asserts that the Hamiltonian  $C$  exhibits a strong form of symmetry protection that extends to all states with energy density above a certain constant threshold.

We shall now argue that for a suitable family of graphs  $G$  all good variational states are qualitatively similar to the GHZ state. Specifically, the results of Refs. [15–17] show that for any constant  $D \geq 3$  there exists an infinite family of bipartite degree- $D$  graphs  $G$  such that

$$C(x) \equiv |\text{cut}(x)| \geq h \min\{|x|, n - |x|\} \quad (4)$$

for any  $x \in \{0, 1\}^n$ , where  $h$  is a constant satisfying

$$h \geq \frac{D}{2} - \sqrt{D-1} \quad (5)$$

and  $|x|$  is the Hamming weight of  $x$ . Such graphs, known as Ramanujan expander graphs, maximize the spectral gap of their adjacency matrices among all  $D$ -regular graphs. Random  $D$ -regular bipartite graphs are known to approach the bound Eq. (5) with high probability [18].

Let  $G$  be a bipartite graph as above and  $x_{\text{opt}} \in \{0, 1\}^n$  be an optimal solution of the MaxCut problem. For a bipartite graph,  $C_{\max} = C(x_{\text{opt}}) = |E|$ . Moreover, the optimal solution  $x_{\text{opt}}$  is unique up to the bitwise negation and

$$C(x) + C(x_{\text{opt}} \oplus x) = |E| \quad (6)$$

for all  $x \in \{0, 1\}^n$ . Here,  $\oplus$  denotes the bitwise XOR. Set  $\epsilon = h/6$  and consider a level- $p$  QAOA state such that

$$\langle \psi(\beta, \gamma) | C | \psi(\beta, \gamma) \rangle \geq |E| - \epsilon n. \quad (7)$$

Let  $x$  be a random  $n$ -bit string sampled from the distribution  $P(x) = |\langle x | \psi(\beta, \gamma) \rangle|^2$ . Markov's inequality and Eq. (7) show that  $C(x) \geq |E| - 2\epsilon n$  with a probability of at least  $1/2$ . From Eq. (6), one infers that  $C(x_{\text{opt}} \oplus x) \leq 2\epsilon n$  with a probability of at least  $1/2$ . Let  $\text{dist}(x, y) = |x \oplus y|$  be the Hamming distance between bit strings  $x$  and  $y$ . Combining Eq. (4) and the bound  $C(x_{\text{opt}} \oplus x) \leq 2\epsilon n$ , one gets

$$\min\{\text{dist}(x_{\text{opt}}, x), \text{dist}(\bar{x}_{\text{opt}}, x)\} \leq \frac{2\epsilon n}{h} = \frac{n}{3} \quad (8)$$

with a probability of at least  $1/2$ . This shows that the state  $|\psi(\beta, \gamma)\rangle$  has a non-negligible weight on bit strings close to

$x_{\text{opt}}$  and on those close to  $\overline{x_{\text{opt}}}$ . Here, closeness means being within a Hamming distance of at most  $n/3$ .

Finally, we employ a fascinating result by Eldar and Harrow stated as Corollary 43 in Ref. [19]. It asserts that the output distribution of a low-depth circuit cannot assign a non-negligible weight to subsets of bit strings that are far apart in Hamming distance. Namely, suppose  $|\psi\rangle$  is an  $n$ -qubit state that can be prepared starting from a product state by a depth- $d$  quantum circuit composed of one- and two-qubit gates. The state  $|\psi\rangle$  does not have to be symmetric in any sense. Define a distribution  $P(x) = |\langle x|\psi\rangle|^2$ . Given a subset  $S \subseteq \{0, 1\}^n$ , let  $P(S) = \sum_{x \in S} P(x)$ . Reference [19] showed that for any subsets  $S, S' \subseteq \{0, 1\}^n$  one has

$$\text{dist}(S, S') \leq \frac{4n^{1/2}2^{3d/2}}{\min\{P(S), P(S')\}}. \quad (9)$$

Here,  $\text{dist}(S, S') = \min_{x \in S} \min_{y \in S'} \text{dist}(x, y)$  is the minimum number of bit flips required to get from  $S$  to  $S'$ . Choose  $S$  and  $S'$  as the sets of  $n$ -bit strings  $x$  such that  $\text{dist}(x_{\text{opt}}, x) \leq n/3$  and  $\text{dist}(x_{\text{opt}}, x) \leq n/3$ , respectively. Then  $\text{dist}(S, S') = n/3$ . Choose  $|\psi\rangle \equiv |\psi(\beta, \gamma)\rangle$ . The  $\mathbb{Z}_2$  symmetry of QAOA states gives  $P(x) = P(\bar{x})$  and, thus,  $P(S) = P(S')$ . We have already shown that  $P(S \cup S') \geq 1/2$  [see Eq. (8)]; that is,  $P(S) = P(S') \geq 1/4$ . Combining this and Eq. (9), one arrives at  $1 \leq 48n^{-1/2}2^{3d/2}$ . This gives a lower bound on the depth  $d$  required to approximate the maximum value  $C_{\text{max}} = |E|$  within a ratio

$$1 - \frac{\epsilon n}{|E|} = 1 - \frac{h}{3D} \leq \frac{5}{6} + \frac{\sqrt{D-1}}{3D}.$$

Here, we recalled that  $\epsilon = h/6$  and  $|E| = Dn/2$  and used Eq. (5). In Appendix A in Supplemental Material [20], we show that the level- $p$  QAOA circuit has depth  $d \leq p(D+1)$  whenever  $G$  is a bipartite degree- $D$  graph. Thus,  $1 \leq 48n^{-1/2}2^{3d/2}$  implies  $p \geq (1/3 \log_2 n - 4)(D+1)^{-1}$ . This concludes the proof of Eq. (3).

The above arguments provide an upper bound on the variational energy for any  $\mathbb{Z}_2$ -symmetric state generated by a low-depth circuit. One may ask if stronger bounds can be derived by making use of the special structure of the QAOA ansatz. Indeed, so far this structure has been used only in establishing the  $\mathbb{Z}_2$  symmetry and expressing the circuit depth  $d$  in terms of the degree  $D$  and the level  $p$ . A notable special feature of the QAOA ansatz is its *geometric locality*—the entangling operators  $e^{i\gamma_a C}$  include interactions only between nearest-neighbor qubits with respect to the underlying graph  $G$ .

To elucidate implications of the geometric locality, consider a toy model known as the *ring of disagrees* [3]. It describes the MaxCut problem on the cycle graph. The latter has a set of vertices  $V = \mathbb{Z}_n = \{0, 1, \dots, n-1\}$ . An edge is drawn between any pair of vertices  $j, k \in \mathbb{Z}_n$

such that  $j = k \pm 1 \pmod{n}$ . Quite recently, Ref. [21] proved that the optimal approximation ratio achieved by the level- $p$  QAOA for the ring of disagrees is bounded above by  $(2p+1)/(2p+2)$  for all  $p$  and conjectured that this bound is tight. Here, we prove a version of this conjecture for arbitrary  $\mathbb{Z}_2$ -symmetric geometrically local variational states. To quantify the geometric locality, let  $\text{dist}(j, k)$  be the distance between qubits  $j$  and  $k$  with respect to the cycle graph  $\mathbb{Z}_n$ . Define the  $R$  neighborhood of a qubit  $j$  as the set  $\{i \in \mathbb{Z}_n : \text{dist}(i, j) \leq R\}$ . A unitary  $U$  acting on  $n$  qubits located at vertices of the cycle graph has range  $R$  if the operator  $U^\dagger Z_j U$  has support on the  $R$  neighborhood of  $j$  for any qubit  $j$ . For example, the level- $p$  QAOA circuit associated with the ring of disagrees has range  $R = p$ . A unitary  $U$  is said to be  $\mathbb{Z}_2$  symmetric if  $UX^{\otimes n} = X^{\otimes n}U$ . Let  $C$  be the MaxCut Hamiltonian Eq. (2) associated with the cycle graph  $\mathbb{Z}_n$ , where  $n$  is even. Note that such a graph has a maximum cut of size  $n$ . We show that

$$\frac{1}{n} \langle +^n | U^\dagger C U | +^n \rangle \leq \frac{2R + 1/2}{2R + 1} \quad (10)$$

for any  $\mathbb{Z}_2$ -symmetric range- $R$  unitary  $U$  with  $R < n/4$ . This bound is tight whenever  $n$  is an even multiple of  $2R + 1$ . Since one can always round  $n$  to the nearest even multiple of  $2R + 1$ , the bound Eq. (10) is tight for all  $n$  up to corrections  $O(1/n)$ , assuming that  $R = O(1)$ .

We shall now prove the upper bound Eq. (10). Let  $\bar{X}$  be the operator that applies Pauli  $X$  to every second qubit, and let  $W = \bar{X}U$ . Note that  $W$  is a  $\mathbb{Z}_2$ -symmetric circuit with range  $R$ . Since  $\bar{X}(Z_j Z_{j+1})\bar{X} = -Z_j Z_{j+1}$  for any qubit  $j$ , the MaxCut Hamiltonian Eq. (2) satisfies  $C + \bar{X}C\bar{X} = nI$ . Taking the expected value of this identity on the state  $U|+^n\rangle$ , one infers that Eq. (10) holds whenever

$$\frac{1}{n} \langle +^n | W^\dagger C W | +^n \rangle \geq 1 - \frac{2R + 1/2}{2R + 1} = \frac{1}{2(2R + 1)}. \quad (11)$$

Thus, it suffices to prove that Eq. (11) holds for any  $\mathbb{Z}_2$ -symmetric range- $R$  circuit  $W$ . For each  $j, k \in \mathbb{Z}_n$  define

$$\epsilon_{j,k} = \frac{1}{2} \langle +^n | W^\dagger (I - Z_j Z_k) W | +^n \rangle.$$

We claim that

$$\epsilon_{j,k} = 1/2 \quad \text{if } \text{dist}(j, k) > 2R. \quad (12)$$

Indeed,  $\langle +^n | W^\dagger Z_j W | +^n \rangle = 0$  for any qubit  $j$ , since the states  $W|+^n\rangle$  and  $Z_j W|+^n\rangle$  are eigenvectors of  $X^{\otimes n}$  with eigenvalues 1 and  $-1$ . Such eigenvectors have to be orthogonal. From  $\text{dist}(j, k) > 2R$ , one infers that  $W^\dagger Z_j W$  and  $W^\dagger Z_k W$  have disjoint support. Thus,

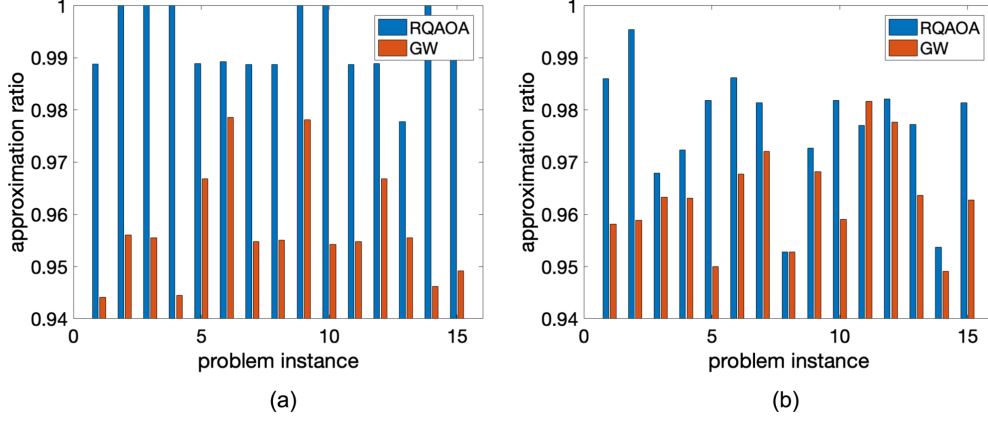


FIG. 1. (a) Approximation ratios achieved by the level-1 RQAOA (blue) and the Goemans-Williamson (GW) algorithm [9] (red) for 15 instances of the Ising cost function with random  $\pm 1$  couplings defined on the 2D toric grid of size  $16 \times 16$ . In case (b), the Ising Hamiltonian also includes random  $\pm 1$  external fields. The RQAOA threshold value is  $n_c = 20$ . We found that the standard level-1 QAOA achieves approximation ratios below  $1/2$  for all considered instances (not shown). The GW algorithm was implemented with  $n = 256$  rounding attempts, and the best found solution was selected. The exact maximum energy was computed using integer linear programming.

$$\begin{aligned} \langle +^n | W^\dagger Z_j Z_k W | +^n \rangle &= \langle +^n | (W^\dagger Z_j W) (W^\dagger Z_k W) | +^n \rangle \\ &= \langle +^n | W^\dagger Z_j W | +^n \rangle \cdot \langle +^n | W^\dagger Z_k W | +^n \rangle \\ &= 0. \end{aligned}$$

This proves Eq. (12).

Suppose one prepares the state  $W|+^n\rangle$  and measures a pair of qubits  $j < k$  in the standard basis. Then  $\epsilon_{j,k}$  is the probability that the measured values on qubits  $j$  and  $k$  disagree. If qubits  $j$  and  $k$  disagree, at least one pair of qubits  $(i, i+1)$  located in the interval  $[j, k]$  must disagree. The probability of the latter event is  $\epsilon_{i,i+1}$ . Thus, the union bound gives

$$\epsilon_{j,k} \leq \sum_{i=j}^{k-1} \epsilon_{i,i+1}. \quad (13)$$

Set  $k = j + 2R + 1$ . Then  $\epsilon_{j,k} = 1/2$  by Eq. (12). Take the expected value of Eq. (13) with respect to random uniform  $j \in \mathbb{Z}_n$ . This gives

$$\frac{1}{2} \leq \frac{2R+1}{n} \sum_{i \in \mathbb{Z}_n} \epsilon_{i,i+1} = \frac{2R+1}{n} \langle +^n | W^\dagger C W | +^n \rangle,$$

proving Eq. (11). As argued above, Eq. (11) is equivalent to Eq. (10). In Appendix B in Supplemental Material [20], we show that the bound Eq. (10) is tight by constructing a  $\mathbb{Z}_2$ -symmetric range- $R$  circuit  $U$  such that  $U|+^n\rangle$  is a tensor product of GHZ-like states on consecutive segments of  $2R+1$  qubits. This circuit is shown to saturate the upper bound Eq. (10).

Motivated by the limitations established above, we propose a nonlocal modification of QAOA which we call the recursive quantum approximate optimization algorithm

(RQAOA). This is a VQO-type algorithm based on the variational ansatz Eq. (1) with a constant level  $p$ . The key new feature of the RQAOA is a variable elimination step. The latter transforms a cost function with  $n$  variables to one with  $n-1$  variables by examining correlations present in the optimal variational state and identifying strongly correlated clusters of variables. To describe this formally, suppose the cost function  $C(x)$  describes the Ising model on a graph  $G = (V, E)$  with  $n$  vertices. The corresponding  $n$ -qubit Hamiltonian is

$$C = \sum_{(j,k) \in E} J_{j,k} Z_j Z_k. \quad (14)$$

Here,  $J_{j,k}$  are arbitrary real coefficients. As before, our goal is to maximize  $C(x) = \langle x | C | x \rangle$ . The RQAOA consists of the following steps.

First, maximize the expected value  $\langle \psi(\beta, \gamma) | C | \psi(\beta, \gamma) \rangle$  over  $\beta, \gamma \in \mathbb{R}^p$ . For every edge  $(j, k) \in E$ , compute the mean value  $M_{j,k} = \langle \psi(\beta, \gamma) | Z_j Z_k | \psi(\beta, \gamma) \rangle$ .

Next, find an edge  $(i, j) \in E$  with the largest magnitude of  $M_{i,j}$  (breaking ties arbitrarily). The corresponding variables  $Z_i$  and  $Z_j$  are correlated if  $M_{i,j} > 0$  and anti-correlated if  $M_{i,j} < 0$ . Impose a parity constraint

$$Z_j = \text{sgn}(M_{i,j}) Z_i \quad (15)$$

and substitute it into the cost function  $C$  to eliminate the variable  $j$ . For example, a term  $J_{j,k} Z_j Z_k$  with  $k \notin \{i, j\}$  gets mapped to  $J_{j,k} \text{sgn}(M_{i,j}) Z_i Z_k$ . The term  $J_{i,j} Z_i Z_j$  gets mapped to a constant energy shift  $J_{i,j} \text{sgn}(M_{i,j})$ . All other terms remain unchanged. This yields a new Ising cost function  $C'$  that depends on  $n-1$  variables. The underlying interaction graph  $G'$  with  $n-1$  vertices is obtained from  $G$

by contracting the edge  $(i, j)$  [22]. The maximum energy of  $C'$  coincides with the maximum energy of  $C$  over the subset of assignments satisfying the constraint Eq. (15).

Next, call the RQAOA recursively to maximize the cost function  $C'$ . Each recursion step eliminates one variable from the cost function. The recursion stops when the number of variables reaches some specified threshold value  $n_c \ll n$ . The remaining instance of the problem with  $n_c$  variables is then solved by a purely classical algorithm (for example, by a brute force method). Thus, the value of  $n_c$  controls how the workload is distributed between quantum and classical computers.

Finally, assign a value to all eliminated variables  $Z_j$  by backtracking the steps of the algorithm and applying the parity constraints Eq. (15) imposed at each step. This results in a tentative solution  $x \in \{0, 1\}^n$  of the original problem with  $n$  variables.

Importantly, the limitations established above for the QAOA with a constant level  $p$  on bounded degree graphs do not apply to its recursive version. Indeed, each variable elimination step performed by the RQAOA results in a contraction of some edge in the graph. The latter tends to increase the maximum vertex degree, thereby increasing the circuit depth of level- $p$  variational states. In other words, the RQAOA overcomes the locality restriction of the standard QAOA without increasing the number of variational parameters that have to be optimized at each step.

We report a numerical comparison between the level-1 RQAOA and the Goemans-Williamson algorithm [9] for the Ising cost function Eq. (14) with random coefficients  $J_{j,k} = \pm 1$ . Two graphs are considered: (a) the 2D grid and (b) the 2D grid with one extra vertex connected to all grid points. The latter is equivalent to the 2D Ising model with random  $\pm 1$  external fields. As shown in Ref. [23], the problem of maximizing the energy  $C(x)$  admits an efficient algorithm in case (a), while case (b) is *NP* hard. To compute the mean values  $\langle \psi(\beta, \gamma) | Z_j Z_k | \psi(\beta, \gamma) \rangle$ , we used a version of the algorithm by Wang *et al.* [24], as detailed in Appendix C in Supplemental Material [20]. Figure 1 shows approximation ratios achieved by each algorithm for 15 problem instances with the grid size  $16 \times 16$ . It can be seen that the RQAOA outperforms the Goemans-Williamson algorithm for all except for one instance. We found that the standard level-1 QAOA achieves an approximation ratio below  $1/2$  for all considered instances. Finally, we show analytically that the RQAOA with the level  $p = 1$  finds the optimal solution for the ring of disagrees model; see Appendix D in Supplemental Material [20]. Meanwhile, the standard level- $p$  QAOA achieves an approximation ratio of at most  $(2p + 1)/(2p + 2)$  for this model [21]. This proves that in certain cases the RQAOA is strictly more powerful than the QAOA.

The authors thank Giacomo Nannicini and Kristan Temme for helpful discussions. S.B. was partially supported by the IBM Research Frontiers Institute and by the

Army Research Office (ARO) under Grant No. W911NF-20-1-0014. E. T. acknowledges the support of the Natural Sciences and Engineering Research Council of Canada (NSERC) and funding provided by the Institute for Quantum Information and Matter, an National Science Foundation (NSF) Physics Frontiers Center (NSF Grant No. PHY-1733907). R. K. and A. K. gratefully acknowledge support by the Deutsche Forschungsgemeinschaft cluster of excellence 2111 (Munich Center for Quantum Science and Technology) and by IBM.

*Note added.*—Recently, analogous limitations were established for random regular graphs by exploiting the locality and spatial uniformity of the QAOA [25,26]. We focus on  $\mathbb{Z}_2$  symmetry and locality, and our statements also apply to nonuniform local algorithms.

- 
- [1] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'Brien, *Nat. Commun.* **5**, 4213 (2014).
  - [2] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta, *Nature (London)* **549**, 242 (2017).
  - [3] E. Farhi, J. Goldstone, and S. Gutmann, [arXiv:1411.4028](https://arxiv.org/abs/1411.4028).
  - [4] A. Y. Kitaev, *Ann. Phys. (Amsterdam)* **303**, 2 (2003).
  - [5] S. Bravyi, M. B. Hastings, and F. Verstraete, *Phys. Rev. Lett.* **97**, 050401 (2006).
  - [6] D. Aharonov and Y. Touati, [arXiv:1810.03912](https://arxiv.org/abs/1810.03912).
  - [7] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, *Nat. Commun.* **9**, 4812 (2018).
  - [8] E. Farhi and A. W. Harrow, [arXiv:1602.07674](https://arxiv.org/abs/1602.07674).
  - [9] M. X. Goemans and D. P. Williamson, *J. ACM* **42**, 1115 (1995).
  - [10] R. Shaydulin and Y. Alexeev, *Evaluating quantum approximate optimization algorithm: A case study*, in *2019 Tenth International Green and Sustainable Computing Conference (IGSC)* (IEEE, New York, 2019), pp. 1–6.
  - [11] M. B. Hastings, [arXiv:1905.07047](https://arxiv.org/abs/1905.07047).
  - [12] Z.-C. Gu and X.-G. Wen, *Phys. Rev. B* **80**, 155131 (2009).
  - [13] N. Schuch, D. Pérez-García, and I. Cirac, *Phys. Rev. B* **84**, 165139 (2011).
  - [14] F. Pollmann, A. M. Turner, E. Berg, and M. Oshikawa, *Phys. Rev. B* **81**, 064439 (2010).
  - [15] A. W. Marcus, D. A. Spielman, and N. Srivastava, Interlacing families iv: Bipartite Ramanujan graphs of all sizes, in *Proceedings of the 2015 IEEE 56th Annual Symposium on Foundations of Computer Science (IEEE, New York, 2015)*, pp. 1358–1377.
  - [16] A. W. Marcus, D. A. Spielman, and N. Srivastava, *Ann. Math.* **182**, 307 (2015).
  - [17] M. Morgenstern, *J. Comb. Theory Ser. B* **62**, 44 (1994).
  - [18] J. Friedman, [arXiv:cs/0405020](https://arxiv.org/abs/cs/0405020).
  - [19] L. Eldar and A. W. Harrow, Local Hamiltonians whose ground states are hard to approximate, in *Proceedings of the 2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)* (IEEE, New York, 2017), pp. 427–438.

- [20] See Supplemental Material at <http://link.aps.org/supplemental/10.1103/PhysRevLett.125.260505> for further details on state preparation and implementation.
- [21] G. Mbeng, R. Fazio, and G. Santoro, [arXiv:1906.08948](https://arxiv.org/abs/1906.08948).
- [22] Recall that contraction of an edge  $(i, j)$  in a graph  $G$  is an operation that removes the edge  $(i, j)$  from  $G$  and identifies the vertices  $i$  and  $j$ .
- [23] F. Barahona, *J. Phys. A* **15**, 3241 (1982).
- [24] Z. Wang, S. Hadfield, Z. Jiang, and E. G. Rieffel, *Phys. Rev. A* **97**, 022304 (2018).
- [25] E. Farhi, D. Gamarnik, and S. Gutmann, [arXiv:2004.09002](https://arxiv.org/abs/2004.09002).
- [26] E. Farhi, D. Gamarnik, and S. Gutmann, [arXiv:2005.08747](https://arxiv.org/abs/2005.08747).

# Obstacles to Variational Quantum Optimization from Symmetry Protection Supplementary Material

Sergey Bravyi <sup>\*</sup>    Alexander Kliesch <sup>†</sup>    Robert Koenig <sup>‡†</sup>    Eugene Tang <sup>§</sup>

September 16, 2020

## A QAOA state preparation circuit

In this section we construct a quantum circuit that prepares the level- $p$  QAOA state for any Ising-type Hamiltonian

$$C = \sum_{(j,k) \in E} J_{j,k} Z_j Z_k$$

defined on a graph  $G = (V, E)$  with  $n$  vertices and maximum vertex degree  $D$ . This includes the MaxCut Hamiltonian as a special case. Let

$$U = \prod_{a=1}^p e^{i\beta_a B} e^{i\gamma_a C}$$

be the requisite circuit. For simplicity, we ignore the initial layer of Hadamard gates that prepares the  $|+\rangle^n$  state.

**Lemma A.1.** *The unitary  $U$  can be realized by a circuit of depth  $d \leq p(D+2)$  composed of 1-qubit and 2-qubit gates. If the graph  $G$  is  $D$ -regular and bipartite then  $d \leq p(D+1)$ .*

*Proof.* By Vizing's theorem [24] there is an edge coloring of  $G$  with at most  $D+1$  colors. Let  $E = E_1 \cup \dots \cup E_{D+1}$  be such a coloring. For each color  $c \in \{1, \dots, D+1\}$  define a unitary

$$V_c = \prod_{(j,k) \in E_c} e^{i\gamma J_{j,k} Z_j Z_k}$$

Note that  $V_c$  is a depth-1 circuit since all edges in  $E_c$  are disjoint. Then each entangling layer  $e^{i\gamma_a C}$  can be realized by a depth  $D+1$  circuit  $V_1 V_2 \dots V_{D+1}$ . Each layer  $e^{i\beta_a B}$  is a product of single-qubit gates, which has depth one. Thus  $U$  has depth at most  $p(D+2)$ .

If  $G$  is  $D$ -regular and bipartite, we may reduce the number of edge colors from  $D+1$  to  $D$  since all bipartite graphs are  $D$ -edge-colorable by König's line coloring theorem. We illustrate the construction of the circuit on Figure 1 for the case  $D=3$  and  $p=1$ . □

---

<sup>\*</sup>IBM T.J. Watson Research Center, Yorktown Heights, NY 10598

<sup>†</sup>Zentrum Mathematik, Technical University of Munich, 85748 Garching, Germany

<sup>‡</sup>Institute for Advanced Study, Technical University of Munich, 85748 Garching, Germany

<sup>§</sup>Institute for Quantum Information and Matter, Caltech, Pasadena, CA 91125



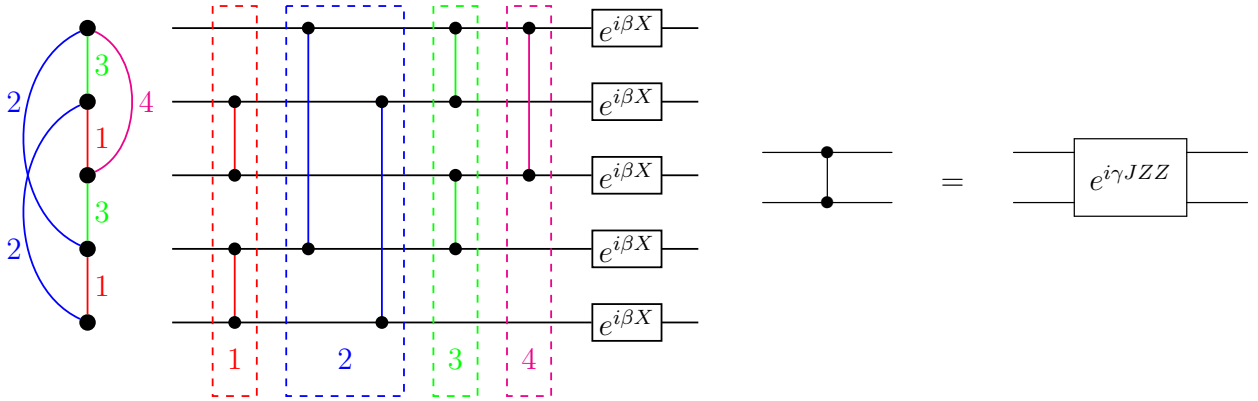


Figure 1: Example for the construction of the circuit given in Lemma A.1: a 4-colorable graph with maximum degree 3 alongside its associated depth-5 quantum circuit for the level-1 QAOA unitary.

## B Optimal variational circuit for the ring of disagrees

In this section we consider the MaxCut Hamiltonian  $C$  on the cycle graph  $\mathbb{Z}_n$ . It is shown that the upper bound

$$\frac{1}{n} \langle +^n | U^\dagger C U | +^n \rangle \leq \frac{2p + 1/2}{2p + 1}.$$

established in the main text for any  $\mathbb{Z}_2$ -symmetric range- $p$  unitary  $U$  with  $p < n/4$  is tight whenever  $n$  is an even multiple of  $2p + 1$ . Let

$$|\text{GHZ}_n\rangle = 2^{-1/2}(|0^n\rangle + |1^n\rangle)$$

be the GHZ state of  $n$  qubits.

**Lemma B.1.** *Suppose  $n = 2p + 1$  for some integer  $p$ . There exists a  $\mathbb{Z}_2$ -symmetric range- $p$  quantum circuit  $V$  such that*

$$|\text{GHZ}_n\rangle = V|+^n\rangle. \quad (1)$$

*Proof.* We shall write  $\text{CX}_{c,t}$  for the CNOT gate with a control qubit  $c$  and a target qubit  $t$ . Let  $H_j$  be the Hadamard gate acting on the  $j$ -th qubit and  $c = p + 1$  be the central qubit. One can easily check that

$$|\text{GHZ}_n\rangle = \left( \prod_{j=1}^p \text{CX}_{c,c-j} \text{CX}_{c,c+j} \right) H_c |0^n\rangle.$$

All CX gates in the product pairwise commute, so the order does not matter. Inserting a pair of Hadamards on every qubit  $j \in [n] \setminus \{c\}$  before and after the respective CX gate and using the identity  $(I \otimes H)\text{CX}(I \otimes H) = \text{CZ}$  one gets

$$|\text{GHZ}_n\rangle = \left( \prod_{j \in [n] \setminus \{c\}} H_j \right) \left( \prod_{j=1}^p \text{CZ}_{c,c-j} \text{CZ}_{c,c+j} \right) |+^n\rangle. \quad (2)$$

Let  $S = \exp[i(\pi/4)Z]$  be the phase-shift gate. Define the two-qubit Clifford gate

$$\text{RZ} = (S \otimes S)^{-1} \text{CZ} = \exp(-i\pi/4) \exp[-i(\pi/4)(Z \otimes Z)].$$

Expressing CZ in terms of RZ and  $S$  in Eq. (2) one gets

$$|\text{GHZ}_n\rangle = S_c^{2p} \left( \prod_{j \in [n] \setminus \{c\}} H_j S_j \right) \left( \prod_{j=1}^p \text{RZ}_{c,c-j} \text{RZ}_{c,c+j} \right) |+\rangle^n. \quad (3)$$

Multiply both sides of Eq. (3) on the left by a product of  $S$  gates over qubits  $j \in [n] \setminus \{c\}$ . Noting that

$$SHS = i \exp[-i(\pi/4)X]$$

one gets (ignoring an overall phase factor)

$$\prod_{j \in [n] \setminus \{c\}} S_j |\text{GHZ}_n\rangle = S_c^{2p} \left( \prod_{j \in [n] \setminus \{c\}} \exp[-i(\pi/4)X_j] \right) \left( \prod_{j=1}^p \text{RZ}_{c,c-j} \text{RZ}_{c,c+j} \right) |+\rangle^n. \quad (4)$$

Using the identity

$$\prod_{j \in [n] \setminus \{c\}} S_j |\text{GHZ}_n\rangle = S_c^{2p} |\text{GHZ}_n\rangle.$$

one can cancel  $S_c^{2p}$  that appears in both sides of Eq. (4). We arrive at Eq. (1) with

$$V = \left( \prod_{j \in [n] \setminus \{c\}} \exp[-i(\pi/4)X_j] \right) \left( \prod_{j=1}^p \text{RZ}_{c,c-j} \text{RZ}_{c,c+j} \right)$$

The circuit diagram of  $V$  in the case  $n = 7$  is shown in Figure 2. Obviously,  $V$  is  $\mathbb{Z}_2$ -symmetric since any individual gate commutes with  $X^{\otimes n}$ . Let us check that  $V$  has range- $p$ . Consider any single-qubit observable  $O_q$  acting on the  $q$ -th qubit. Consider three cases. *Case 1:*  $q = c$ . Then  $V^\dagger O_q V$  may be supported on all  $n$  qubits. However,  $[c-p, c+p] = [1, n]$ , so the  $p$ -range condition is satisfied trivially. *Case 2:*  $1 \leq q < c$ . Then all gates  $\text{RZ}_{c,c+j}$  in  $V$  cancel the corresponding gates in  $V^\dagger$ , so that  $V^\dagger O_q V$  has support in the interval  $[1, c] \subseteq [q-p, q+p]$ . Thus the  $p$ -range condition is satisfied. *Case 3:*  $c < q \leq n$ . This case is equivalent to Case 2 by symmetry.  $\square$

Recall that the ring of disagrees Hamiltonian has the form

$$C = \frac{1}{2} \sum_{j \in \mathbb{Z}_n} (I - Z_j Z_{j+1}).$$

**Lemma B.2.** *Consider any integers  $n, p$  such that  $n$  is even and  $n$  is a multiple of  $2p + 1$ . Then there exists a  $\mathbb{Z}_2$ -symmetric range- $p$  circuit  $U$  such that*

$$\frac{1}{n} \langle +^n | U^\dagger C U | +^n \rangle = \frac{2p + 1/2}{2p + 1}.$$

*Proof.* Let  $W$  be the  $\mathbb{Z}_2$ -symmetric range- $p$  unitary operator preparing the GHZ state on  $2p + 1$  qubits starting from  $|+^{2p+1}\rangle$ , see Lemma B.1. Suppose  $n = m(2p + 1)$  for some even integer  $m$ . Define

$$U = \bar{X} W^{\otimes m},$$

where

$$\bar{X} = (X \otimes I)^{\otimes n/2}.$$

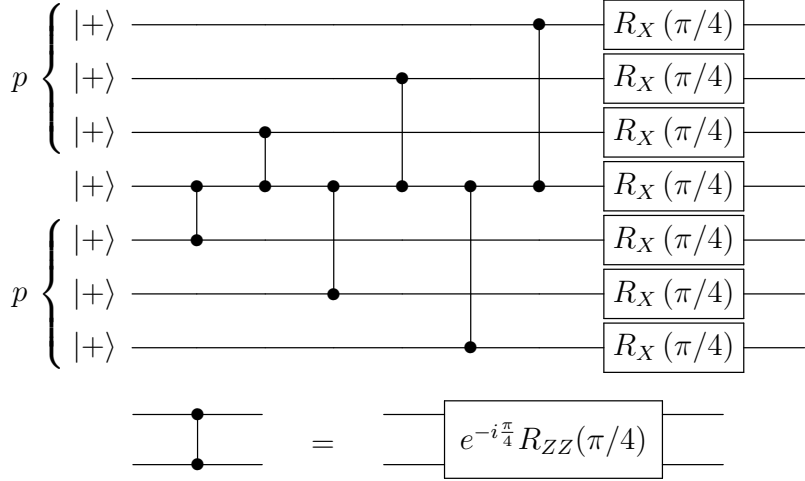


Figure 2: The  $\mathbb{Z}_2$ -symmetric range-3 quantum circuit to prepare the GHZ state  $|\text{GHZ}_{2p+1}\rangle$  of  $2p+1 = 7$  qubits ( $p = 3$ ). Here,  $R_O(\theta) = \exp(-i\theta O)$ .

Since each copy of  $W$  acts on a consecutive interval of qubits and has range  $p$ , one infers that  $U$  has range  $p$ . We have

$$\bar{X}^\dagger C \bar{X} = \sum_{k \in \mathbb{Z}_n} G_k, \quad \text{where} \quad G_k = \frac{1}{2}(I + Z_k Z_{k+1}).$$

The state  $W^{\otimes m}|+^n\rangle$  is a tensor product of GHZ states supported on consecutive tuples of  $2p + 1$  qubits. The expected value of  $G_k$  on the state  $W^{\otimes m}|+^n\rangle$  equals 1 if  $G_k$  is supported on one of the GHZ states. Otherwise, if  $G_k$  crosses the boundary between two GHZ states, the expected value of  $G_k$  on the state  $W^{\otimes m}|+^n\rangle$  equals  $1/2$ . Thus

$$\langle +^n | U^\dagger C U | +^n \rangle = \sum_{k \in \mathbb{Z}_n} \langle +^n | (W^{\otimes m})^\dagger G_k W^{\otimes m} | +^n \rangle = m(2p + 1/2) = n \left( \frac{2p + 1/2}{2p + 1} \right).$$

□

## C Numerical simulation of level-1 QAOA and RQAOA

In this section we provide details of the simulation reported on Figure 1 in the main text. Let  $J$  be a real symmetric matrix of size  $n$ . Consider an Ising-type Hamiltonian

$$C = \sum_{1 \leq j < k \leq n} J_{j,k} Z_j Z_k.$$

Here  $J_{j,k}$  are arbitrary real coefficients. Below we show how to compute the mean value of a Pauli operator  $Z_j Z_k$  on the level-1 QAOA state

$$|\psi(\beta, \gamma)\rangle = e^{i\beta B} e^{i\gamma C} | +^n \rangle$$

in time  $O(n)$  using an explicit analytic formula. Such a formula was derived for the MaxCut cost function by Wang et al. [27, Theorem 1]. Here we provide a generalization to arbitrary Ising

Hamiltonians. Since the total number of terms in the cost function is  $O(n^2)$ , simulating each step of RQAOA takes time at most  $O(n^3)$ . Assuming that  $n_c = O(1)$ , the number of steps is roughly  $n$  so that the full simulation cost is  $O(n^4)$ . Crucially, the simulation cost of this method does not depend on the depth of the variational circuit. This is important because RQAOA may potentially increase the depth from  $O(1)$  to  $O(n)$  since it adds many new terms to the cost function.

**Lemma C.1.** *Fix a pair of qubits  $1 \leq j < k \leq n$ . Let  $c = \cos(2\beta)$  and  $s = \sin(2\beta)$ . Then*

$$\begin{aligned} \langle \psi(\beta, 1) | Z_j Z_k | \psi(\beta, 1) \rangle &= (s^2/2) \prod_{p \neq j, k} \cos[2J_{j,p} - 2J_{k,p}] - (s^2/2) \prod_{p \neq j, k} \cos[2J_{j,p} + 2J_{k,p}] \\ &\quad + cs \cdot \sin(2J_{j,k}) \left[ \prod_{p \neq j, k} \cos(2J_{j,p}) + \prod_{p \neq j, k} \cos(2J_{k,p}) \right]. \end{aligned} \quad (5)$$

Here we only consider the case  $\gamma = 1$  since  $\gamma$  can be absorbed into the definition of  $J$ .

*Proof.* Given a 2-qubit observable  $O$  define the mean value

$$\mu(O) = \langle \psi(\beta, 1) | O_{j,k} | \psi(\beta, 1) \rangle.$$

We are interested in the observable  $O = ZZ \equiv Z \otimes Z$ .

We note that all terms in  $B$  and  $C$  that act trivially on  $\{j, k\}$  do not contribute to  $\mu(O)$ . Such terms can be set to zero. Given a 2-qubit observable  $O$ , define a mean value

$$\mu'(O) = \langle +^n | e^{iC'} O_{j,k} e^{-iC'} | +^n \rangle, \quad \text{where} \quad C' = \sum_{p \neq j, k} (J_{j,p} Z_j + J_{k,p} Z_k) Z_p. \quad (6)$$

Using the identities

$$\begin{aligned} e^{i\beta(X_j + X_k)} Z_j Z_k e^{-i\beta(X_j + X_k)} &= c^2 Z_j Z_k + s^2 Y_j Y_k + cs(Z_j Y_k + Y_j Z_k), \\ e^{iJ_{j,k} Z_j Z_k} Z_j Z_k e^{-iJ_{j,k} Z_j Z_k} &= Z_j Z_k, \\ e^{iJ_{j,k} Z_j Z_k} Y_j Y_k e^{-iJ_{j,k} Z_j Z_k} &= Y_j Y_k \\ e^{iJ_{j,k} Z_j Z_k} Z_j Y_k e^{-iJ_{j,k} Z_j Z_k} &= \cos(2J_{j,k}) Z_j Y_k + \sin(2J_{j,k}) X_k, \\ e^{iJ_{j,k} Z_j Z_k} Y_j Z_k e^{-iJ_{j,k} Z_j Z_k} &= \cos(2J_{j,k}) Y_j Z_k + \sin(2J_{j,k}) X_j, \end{aligned}$$

and noting that  $\mu'(ZZ) = 0$  one easily gets

$$\mu(ZZ) = s^2 \cdot \mu'(YY) + cs \cdot \cos(2J_{j,k}) [\mu'(ZY) + \mu'(YZ)] + cs \cdot \sin(2J_{j,k}) [\mu'(XI) + \mu'(IX)]. \quad (8)$$

Using the explicit form of  $C'$  one gets

$$e^{-iC'} | +^n \rangle = \frac{1}{2} \sum_{a, b=0,1} |a, b\rangle_{j,k} \otimes |\Phi(a, b)\rangle_{\text{else}}, \quad (9)$$

where  $|\Phi(a, b)\rangle$  is a tensor product state of  $n - 2$  qubits defined by

$$|\Phi(a, b)\rangle = \bigotimes_{p \neq j, k} |J_{j,p}(-1)^a + J_{k,p}(-1)^b\rangle_p \quad \text{where} \quad |\theta\rangle \equiv e^{-i\theta Z} | + \rangle.$$

Combining Eqs. (6),(9) one gets

$$\mu'(O) = (1/4) \sum_{a, b, a', b'=0,1} \langle a', b' | O | a, b \rangle \cdot \langle \Phi(a', b') | \Phi(a, b) \rangle. \quad (10)$$

Using the tensor product form of the states  $|\Phi(a, b)\rangle$  and the identity  $\langle\theta'|\theta\rangle = \cos(\theta - \theta')$  gives

$$\langle\Phi(a', b')|\Phi(a, b)\rangle = \prod_{p \neq j, k} \cos [J_{j,p}(-1)^a - J_{j,p}(-1)^{a'} + J_{k,p}(-1)^b - J_{k,p}(-1)^{b'}]. \quad (11)$$

From Eqs. (10),(11) one can easily compute the mean value  $\mu'(O)$  for any 2-qubit observable.

Consider first the case  $O = YY$ . Then the only terms contributing to Eq. (10) are those with  $a' = a \oplus 1$  and  $b' = b \oplus 1$ . The identity  $\langle a \oplus 1 | Y | a \rangle = -i(-1)^a$  gives

$$\mu'(YY) = -(1/4) \sum_{a,b=0,1} (-1)^{a+b} \prod_{p \neq j, k} \cos [2J_{j,p}(-1)^a + 2J_{k,p}(-1)^b],$$

that is,

$$\mu'(YY) = (1/2) \prod_{p \neq j, k} \cos [2J_{j,p} - 2J_{k,p}] - (1/2) \prod_{p \neq j, k} \cos [2J_{j,p} + 2J_{k,p}]. \quad (12)$$

Next consider the case  $O = YZ$ . Note that the matrix elements  $\langle a', b' | O | a, b \rangle$  have zero real part. From Eqs. (10),(11) one infers that  $\mu'(YZ)$  has zero real part. This implies

$$\mu'(YZ) = \mu'(ZY) = 0. \quad (13)$$

Finally, consider the case  $O = XI$ . Then the only terms that contribute to Eq. (10) are those with  $a' = a \oplus 1$  and  $b' = b$ . We get

$$\mu'(XI) = \prod_{p \neq j, k} \cos (2J_{j,p}). \quad (14)$$

Here we noted that the inner product Eq. (11) with  $a' = a \oplus 1$  and  $b' = b$  does not depend on  $a, b$ . By the same argument,

$$\mu'(IX) = \prod_{p \neq j, k} \cos (2J_{k,p}). \quad (15)$$

Combining Eq. (8) and Eqs. (12),(13),(14),(15) one arrives at Eq. (5).  $\square$

Clearly, the ability to simulate level-1 RQAOA with Ising-type cost functions on a classical computer in polynomial time precludes exponential quantum speedups. However, as far as we know, higher-level RQAOA with  $p \geq 2$  lacks efficient classical simulation leaving room for a quantum advantage.

## D RQAOA optimally solves the ring of disagrees

In this section we prove that the level-1 RQAOA optimally solves the ring of disagrees model. This is in sharp contrast to the standard QAOA which achieves approximation ratio at most  $(2p+1)/(2p+2)$  for any level  $p$ , as was shown in Ref. [22]. More generally, we show that the level-1 RQAOA optimally solves any 1D Ising model where the coupling coefficients are either  $+1$  or  $-1$ .

**Lemma D.1.** *Consider a cost function*

$$C(x) = \sum_{k \in \mathbb{Z}_n} J_k (-1)^{x_k + x_{k+1}}$$

with  $n$  variables  $x \in \{0, 1\}^n$  located at vertices of the cycle graph  $\mathbb{Z}_n$ . Assume that  $J_k \in \{1, -1\}$  for all  $k \in \mathbb{Z}_n$ . Then the level-1 RQAOA outputs  $x^* \in \{0, 1\}^n$  such that  $C(x^*) = \max_x C(x)$ .

*Proof.* Let

$$C = \sum_{k \in \mathbb{Z}_n} J_k Z_k Z_{k+1} \quad (16)$$

be the corresponding Hamiltonian. First, we observe that  $\langle \psi(\beta, \gamma) | Z_i Z_j | \psi(\beta, \gamma) \rangle = 0$  if  $\text{dist}(i, j) > 2$  since in this case the operators  $U^{-1} Z_i U$  and  $U^{-1} Z_j U$  have disjoint support. Lemma C.1 shows that

$$\langle \psi(\beta, \gamma) | Z_i Z_j | \psi(\beta, \gamma) \rangle = \begin{cases} \frac{1}{2} J_i \sin(4\beta) \sin(4\gamma) & \text{if } j = i + 1 \\ \frac{1}{4} J_i J_{i+1} \sin^2(2\beta) \sin^2(4\gamma) & \text{if } j = i + 2 \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

when  $J_k \in \{1, -1\}$  for every  $k \in \mathbb{Z}_n$ . Here we assumed  $i < j$ . Thus

$$|\langle \psi(\beta, \gamma) | Z_i Z_{i+2} | \psi(\beta, \gamma) \rangle| \leq 1/4 \quad (18)$$

for all  $\beta, \gamma$ . Let  $\beta^*, \gamma^*$  be the optimal angles maximizing the variational energy  $\langle \psi(\beta, \gamma) | C | \psi(\beta, \gamma) \rangle$ . Then we can infer from Eq. (17) that

$$\langle \psi(\beta^*, \gamma^*) | Z_i Z_{i+1} | \psi(\beta^*, \gamma^*) \rangle = J_i / 2. \quad (19)$$

Combined with Eq. (17) and Eq. (18) we conclude that the maximally correlated pair of variables are nearest neighbors, that is,

$$\arg \max_{(i,j): i < j} |\langle \psi(\beta^*, \gamma^*) | Z_i Z_j | \psi(\beta^*, \gamma^*) \rangle| = (i^*, i^* + 1) \quad (20)$$

for some  $i^* \in \mathbb{Z}_n$ . Without loss of generality, assume that  $i^* = n - 2$ . Then, according to Eq. (20), the RQAOA algorithm eliminates the variable  $Z_{n-1}$ . By Eq. (19), the corresponding parity constraint is

$$Z_{n-1} = Z_{n-2} J_{n-2}. \quad (21)$$

The resulting reduced graph obtained from  $\mathbb{Z}_n$  by contracting the edge  $(n - 1, n - 2)$  is isomorphic to  $\mathbb{Z}_{n-1}$ . It is easy to check that the new cost function Hamiltonian  $C'$  acting on  $n - 1$  qubits is

$$C' = 1 + \sum_{k \in \mathbb{Z}_{n-1}} J'_k Z_k Z_{k+1} \quad (22)$$

with

$$J'_i = \begin{cases} J_i & \text{if } i \neq n - 2 \\ J_{n-2} J_{n-1} & \text{if } i = n - 2 \end{cases} \quad (23)$$

We note that the transformation Eq. (23) preserves the parity of the couplings in the sense that

$$\prod_{k \in \mathbb{Z}_n} J_k = \prod_{k \in \mathbb{Z}_{n-1}} J'_k. \quad (24)$$

Proceeding inductively, one eliminates variables  $Z_{n-1}, Z_{n-2}, \dots, Z_{n_c}$  while imposing parity constraints (cf. Eq. (21))

$$\begin{aligned} Z_{n-1} &= Z_{n-2} J_{n-2} \\ Z_{n-2} &= Z_{n-3} J'_{n-3} \\ &\vdots \end{aligned}$$

arriving at the cost function Hamiltonian  $C''$  for an Ising chain of length  $n_c$  having couplings  $\pm 1$ . Because of Eq. (24) and because the Hamiltonian Eq. (16) is frustrated if and only if  $\prod_{k \in \mathbb{Z}_n} J_k = -1$ , we conclude that any maximum  $x^* \in \{0, 1\}^{n_c}$  of  $C''(x)$  satisfies

$$C''(x^*) = \begin{cases} n_c & \text{if } \prod_{k \in \mathbb{Z}_{n_c}} J_k = 1 \\ n_c - 2 & \text{if } \prod_{k \in \mathbb{Z}_{n_c}} J_k = -1. \end{cases}$$

Because the cost function acquires a constant energy shift in every variable elimination, see Eq. (22), the final output  $x$  of the RQAOA algorithm satisfies

$$C(x) = n - n_c + C''(x^*) = \begin{cases} n & \text{if } \prod_{k \in \mathbb{Z}_n} J_k = 1 \\ n - 2 & \text{if } \prod_{k \in \mathbb{Z}_n} J_k = -1. \end{cases}$$

This implies the claim. □

## References

- [1] C.-K. Chiu, J. C. Y. Teo, A. P. Schnyder, and S. Ryu, *Rev. Mod. Phys.* **88**, 035005 (2016).
- [2] X. Chen, Z.-C. Gu, and X.-G. Wen, *Phys. Rev. B* **82**, 155138 (2010).
- [3] A. Y. Kitaev, *Annals of Physics* **303**, 2 (2003).
- [4] S. Bravyi, M. B. Hastings, and F. Verstraete, *Phys. Rev. Lett.* **97**, 050401 (2006).
- [5] D. Aharonov and Y. Touati, preprint (2018), arXiv:1810.03912.
- [6] M. H. Freedman and M. B. Hastings, *Quantum Information and Computation* **14** (2014).
- [7] M. B. Hastings, *Quantum Information and Computation* **13**, 393 (2013).
- [8] L. Eldar and A. W. Harrow, Local Hamiltonians whose ground states are hard to approximate, in *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 427–438, 2017.
- [9] D. Aharonov, I. Arad, and T. Vidick, *SIGACT News* **44**, 47 (2013).
- [10] C. Nirkhe, U. Vazirani, and H. Yuen, Approximate Low-Weight Check Codes and Circuit Lower Bounds for Noisy Ground States, in *Proceedings of ICALP 2018*, edited by I. C. et al., , Leibniz International Proceedings in Informatics (LIPIcs) Vol. 107, pp. 91:1–91:11, Dagstuhl, Germany, 2018, Dagstuhl–Leibniz-Zentrum fuer Informatik.
- [11] Z.-C. Gu and X.-G. Wen, *Phys. Rev. B* **80**, 155131 (2009).

- [12] F. D. M. Haldane, *Phys. Rev. Lett.* **50**, 1153 (1983).
- [13] I. Affleck, T. Kennedy, E. H. Lieb, and H. Tasaki, *Phys. Rev. Lett.* **59**, 799 (1987).
- [14] N. Schuch, D. Pérez-García, and I. Cirac, *Phys. Rev. B* **84**, 165139 (2011).
- [15] F. Pollmann, A. M. Turner, E. Berg, and M. Oshikawa, *Phys. Rev. B* **81**, 064439 (2010).
- [16] T.-C. Wei, I. Affleck, and R. Raussendorf, *Phys. Rev. A* **86**, 032328 (2012).
- [17] J. Miller and A. Miyake, *Phys. Rev. Lett.* **114**, 120506 (2015).
- [18] M. Morgenstern, *Journal of Combinatorial Theory, Series B* **62**, 44 (1994).
- [19] E. Farhi, J. Goldstone, and S. Gutmann, preprint (2014), arXiv:1411.4028.
- [20] M. B. Hastings, preprint (2019), arXiv:1905.07047.
- [21] M. X. Goemans and D. P. Williamson, *J. ACM* **42**, 1115 (1995).
- [22] G. Mbeng, R. Fazio, and G. Santoro, preprint (2019), arXiv:1906.08948.
- [23] P. Raghavan and C. D. Tompson, *Combinatorica* **7**, 365, (1987).
- [24] V. G. Vizing, *Diskret. Analiz* **3**, 25 (1964).
- [25] A. W. Marcus, D. A. Spielman, and N. Srivastava, Interlacing families iv: Bipartite Ramanujan graphs of all sizes, in *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pp. 1358–1377, 2015.
- [26] A. W. Marcus, D. A. Spielman, and N. Srivastava, *Annals of Mathematics* **182**, 307 (2015).
- [27] Z. Wang, S. Hadfield, Z. Jiang, and E. G. Rieffel, *Phys. Rev. A* **97**, 022304 (2018).
- [28] M. Van Den Nest, *Quantum Info. Comput.* **11**, 784 (2011).
- [29] S. Bravyi *et al.*, *Quantum* **3**, 181 (2019).
- [30] E. Farhi and A. W. Harrow, preprint (2016), arXiv:1602.07674.